AAMRL-TR-89-048

AD-A225 582

# MAPPING MISSIONS ONTO $C^3$ ORGANIZATIONS: INCORPORATING THE GOAL DIMENSION IN IAT

Petros Kapâsouris
Daniel Serfaty
Joseph G. Wohl
James C. Deckert
Krishna R. Pattipati

ALPHATECH, INC.



MAY 1990

FINAL REPORT FOR PERIOD OCTOBER 1987 - OCTOBER 1988



HARRY G. ARMSTRONG AEROSPACE MEDICAL RESEARCH LABORATORY
HUMAN SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6573

## TECHNICAL REVIEW AND APPROVAL

AAMRL-TR-89-048

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

**FOR THE COMMANDER**

**CHARLES BATES, JR.**
Director, Human Engineering Division
Armstrong Aerospace Medical Research Laboratory

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

# REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

| | |
|---|---|
| 1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED | 1b. RESTRICTIVE MARKINGS |
| 2a. SECURITY CLASSIFICATION AUTHORITY | 3. DISTRIBUTION / AVAILABILITY OF REPORT: Approved for public release; distribution is unlimited. |
| 2b. DECLASSIFICATION / DOWNGRADING SCHEDULE | |
| 4. PERFORMING ORGANIZATION REPORT NUMBER(S): TR-414-1 | 5. MONITORING ORGANIZATION REPORT NUMBER(S): AAMRL-TR-89-048 |
| 6a. NAME OF PERFORMING ORGANIZATION: ALPHATECH, Inc. — 6b. OFFICE SYMBOL (If applicable) | 7a. NAME OF MONITORING ORGANIZATION: Harry G. Armstrong Aerospace Medical Research Laboratory - AAMRL/HED |
| 6c. ADDRESS (City, State, and ZIP Code): 50 Mall Road, Burlington, MA 01803-4901 | 7b. ADDRESS (City, State, and ZIP Code): Wright Patterson AFB, OH 45433-6573 |
| 8a. NAME OF FUNDING / SPONSORING ORGANIZATION — 8b. OFFICE SYMBOL (If applicable) | 9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: F33615-86-C-0345 |
| 8c. ADDRESS (City, State, and ZIP Code) | 10. SOURCE OF FUNDING NUMBERS |

| PROGRAM ELEMENT NO. | PROJECT NO. | TASK NO. | WORK UNIT ACCESSION NO. |
|---|---|---|---|
| 62202F | 6893 | 04 | 63 |

11. TITLE (Include Security Classification)

Mapping Missions Onto $C^3$ Organizations: Incorporating the Goal Dimension in IAT (U)

12. PERSONAL AUTHOR(S)

Petros Kapasouris, Daniel Serfaty, Joseph G. Wohl, James C. Deckert, Krishna R. Pattipati (Univ. of Conn.)

| 13a. TYPE OF REPORT | 13b. TIME COVERED | 14. DATE OF REPORT (Year, Month, Day) | 15. PAGE COUNT |
|---|---|---|---|
| FINAL | FROM 10/87 TO 10/88 | 1990 May 09 | 49 |

16. SUPPLEMENTARY NOTATION

| 17. COSATI CODES | | | 18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number) |
|---|---|---|---|
| FIELD | GROUP | SUB-GROUP | optimal process mapping, system representation, system analysis, performance analysis, Petri nets, queuing networks |
| 12 | 04 | | |
| 25 | 05 | | |

19. ABSTRACT (Continue on reverse if necessary and identify by block number)


Previous work on Integrated Analysis Techniques (IAT) for $C^3$ system representation, analysis, and design developed a representation framework consisting of the four dimensions of process, resource, organization, and goal; introduced stochastic, timed, attributed Petri nets (STAPNs) as an appropriate mathematical modeling construct; and developed tools and techniques to incorporate three of the IAT dimensions (process, resource, and organization) into that construct.

In this report we address the remaining goal dimension of IAT. Specifically, we view the goal of the system as consisting of subgoals associated with processes performed by resources within the system. The goal for each process has two subgoals: accuracy and timeliness. The accuracy component for each process is met by stipulating the amount of time the various capable resources will require to complete the process leaving the remaining goal of timeliness free. To address this, we introduce Mapper, a prototype tool that maps the processes onto the capable resources in such a way as to minimize the completion time of the last process in a mission, and also illustrate a modeling and sensitivity analysis technique using Petri nets.


| | |
|---|---|
| 20. DISTRIBUTION / AVAILABILITY OF ABSTRACT: [X] UNCLASSIFIED/UNLIMITED [ ] SAME AS RPT. [ ] DTIC USERS | 21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED |
| 22a. NAME OF RESPONSIBLE INDIVIDUAL: Maj. Michael McFarren | 22b. TELEPHONE (Include Area Code): (513) 255-8822 — 22c. OFFICE SYMBOL: AAMRL/HED |

DD Form 1473, JUN 86 — Previous editions are obsolete — SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

## SUMMARY

Previous work on Integrated Analysis Techniques (IAT) for $C^3$ system representation, analysis, and design developed a representation framework consisting of the four dimensions of process, resource, organization, and goal; introduced stochastic, timed, attributed Petri nets (STAPNs) as an appropriate mathematical modeling construct; and developed tools and techniques to incorporate three of the IAT dimensions (process, resource, and organization) into that construct.

In this report we address the remaining goal dimension of IAT. Specifically, we view the goal of the system as consisting of subgoals associated with processes performed by resources within the system. The goal for each process has two subgoals: accuracy and timeliness. The accuracy component for each process is met by stipulating the amount of time the various capable resources will require to complete the process, leaving the remaining goal of timeliness free. To address this, we introduce Mapper, a prototype tool that maps the processes onto the capable resources in such a way as to minimize the completion time of the last process in a mission, and also illustrate a modeling and sensitivity analysis technique using Petri nets.

## PREFACE

This work was conducted by personnel of ALPHATECH, Inc., under contract F33615-86-C-0545 with the Harry G. Armstrong Aerospace Medical Research Laboratory, Human Systems Division, Air Force Systems Command, Wright Patterson Air Force Base, Ohio. The methods summarized in this report were developed under program 62202F, Aerospace Biotechnology, Project 7184, Man-Machine Integration Technology.

The authors would like to thank Mr. Donald Monk and Major Michael McFarren of AAMRL for their continuing encouragement and support in the face of sometimes seemingly insurmountable obstacles.

# CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

## SECTION 1

## INTRODUCTION AND PROBLEM DEFINITION

### 1.1 INTRODUCTION

In previous work on the development of Integrated Analysis Techniques (IAT) for the analysis and design of $C^3$ systems [1], we developed a framework for the representation and analysis of these systems consisting of:

- A hierarchical method for describing a $C^3$ system along the four dimensions of process, resource, organization, and goal, and
- A mathematical construct for $C^3$ system modeling and performance analysis: stochastic, timed, attributed Petri nets (STAPNs).

We subsequently identified the need for a user-friendly, highly automated tool for $C^3$ system analysis and design, and developed a specification for one form of the tool [4]. More recently, we developed a PC-based tool to support the decomposition of the process and resource dimensions of IAT [3], and demonstrated a technique for representing the organizational dimension via STAPNs[4].

In this report we address the remaining goal dimension of IAT. Specifically, we view the goal of the system as consisting of subgoals associated with processes performed by resources within the system. The goal for each process has two components: accuracy and timeliness. The accuracy component for each process is met by stipulating the amount of time the various capable resources will require to complete the process, leaving the remaining goal of timeliness free. In the sequel we introduce a prototype tool, Mapper, that maps the processes onto the capable resources in such a way as to minimize the completion time of the last process in a mission, and also illustrate a modeling and sensitivity analysis technique using Petri nets.

## 1.2 DEFINITIONS

A military mission may be defined as: 1) a set of military objectives or goals to be attained (e.g., defend a geographic sector); and 2) specification of the organizational elements or units (e.g., regiments, task forces) that will attain the mission goals. Each organizational element has certain limited resources available to it (e.g., humans, computers, ships, aircraft, tanks, artillery, communications gear), and is trained to carry out selected military functions, tasks, or processes (e.g., situation assessment, planning, anti-terrorist activities, mine-laying, precision bombing).

In an earlier report [1] we demonstrated that analysis and modeling of $C^3$ systems requires these four separate but related dimensions of description. Minimally, we define them as follows:

- Resource: a physical mechanism, a human, a geographic location, or a node
- Process: an automated function, a human task, a procedure, or an algorithm
- Organizational element: a subdivision, a unit, or an individual
- Goal: a performance objective, or an intended result

The requirement for aggregation, along with its dual requirement for decomposition, was found to apply equally well to all four dimensions. Thus, an entire trained and equipped armored division with all its tanks, artillery, personnel, and armored personnel carriers may be considered as a single resource by a high-level headquarters; the process of air defense of an area may involve thousands of sub-processes (i.e., functions or tasks), some carried out by humans and others by computers, radars, missiles and aircraft; and so on. In addition, the level of descriptive detail required for $C^3$ system modeling and analysis necessitates the ability to decompose a system along all four dimensions in a consistent manner, while retaining the critical interrelationships among them.

At any level L in the $C^3$ system decomposition, it is necessary that:

- A process description contain references to: 1) the input and output functional dependencies between itself and those other processes at the same level of

decomposition with which it is directly related; 2) the resources required for its performance; 3) the organizational element responsible for monitoring and/or controlling the process; and 4) the goal (i.e., performance requirement) that the process must meet.

- A resource description contain references to: 1) the physical connectivities between itself and those other resources at the same decomposition level required to support the process(es) to which it is assigned; 2) the process(es) which that resource is assigned to support; and 3) the organizational element responsible for the resource.

- An organizational element description contain references to: 1) the lines of authority, responsibility and coordination between itself and those other organizational elements both at the same and at higher and lower decomposition levels, as required to attain the assigned goal(s); 2) the goal(s) for which it is responsible; 3) the resources assigned to it; and 4) the processes which it is responsible for controlling and/or monitoring.

- A goal description contain references to: 1) the higher-level goal of which it is a part, as well as the lower-level goals which must be met in the interests of its own attainment; 2) the organizational element responsible for its attainment; and 3) the process(es) for which that goal is a performance requirement.

This cross-referencing is accomplished by means of assignment matrices. A set of possible relationships among the four dimensions is shown in Table 1-1.

TABLE 1-1. RELATIONSHIPS AMONG THE FOUR DIMENSIONS

| | | REPRESENTATION | | | |
|---|---|---|---|---|---|
| DESCRIPTOR | DEFINITION | PROCESS | RESOURCE | ORG'L ELT | GOAL |
| ISA | Is known as | (name) | (name) | (name) | (name) |
| AKA | Also known as | (name) | (name) | (name) | (name) |
| POF | Part of | $P^L{}_i \in P^{L-1}{}_j$ | $R^L{}_i \in R^{L-1}{}_j$ | $O^L{}_i \in O^{L-1}{}_j$ | $G^L{}_i \in G^{L-1}{}_j$ |
| COF | Consists of | $P^L{}_i=\{P^{L+1}{}_j\}$ | $R^L{}_i=\{R^{L+1}{}_j\}$ | $O^L{}_i=\{O^{L+1}{}_j\}$ | $G^L{}_i=\{G^{L+1}{}_j\}$ |
| STO | Sends to; etc. | $[P^L{}_i \times P^L{}_j]$ | $[R^L{}_i \times R^L{}_j]$ | $[O^L{}_i \times O^L{}_j]$ | ----------- |
| RFM | Rcvs from;etc. | $[P^L{}_i \times P^L{}_j]$ | $[R^L{}_i \times R^L{}_j]$ | $[O^L{}_i \times O^L{}_j]$ | ----------- |
| ATO | Assigned to | $[R^L{}_i \times P^L{}_j]$ | $[R^L{}_i \times O^L{}_j]$ | $[O^{L-1}{}_i \times O^L{}_j]$ | $[G^L{}_i \times P^L{}_j]$ |
| | | $[P^L{}_i \times O^L{}_j]$ | ----------- | ------------ | $[G^L{}_i \times O^L{}_j]$ |
| AST | Assignable to | $[R^L{}_i \times P^L{}_j]^*$ | $[R^L{}_i \times O^L{}_j]^*$ | $[O^{L-1}{}_i \times O^L{}_j]^*$ | $[G^L{}_i \times P^L{}_j]^*$ |
| | | $[P^L{}_i \times O^L{}_j]^*$ | ----------- | ------------ | $[G^L{}_i \times O^L{}_j]^*$ |

Notes:
- Superscript = level of decomposition
- Subscript = index
- Read $R^L{}_i$ as "i-th resource at level L"
- Read [A x B] as "A sends to etc.; receives from etc.; or is assigned to B"
- Read [A x B]* as "A is assignable to B"
- Read $G^L{}_i \in G^{L-1}{}_j$ as "goal i at level L is a part of goal j at level L-1

## 1.3 PROBLEM STATEMENT

A hypothetical analysis tool for iteratively defining an organizational structure and allocating sequences of processes to that structure might appear as shown in Fig. 1-1. The "mission generator" creates a set of possible missions. Then we assume an organizational structure with its structural constraints (communication constraints, hierarchical constraints, etc.) and we assign all the generated missions to that organizational structure. With a set of performance criteria we evaluate the assignments, we change the organizational structure and reassign the missions to the new organizational structure. We repeat the process until the organizational structure meets the performance criteria.

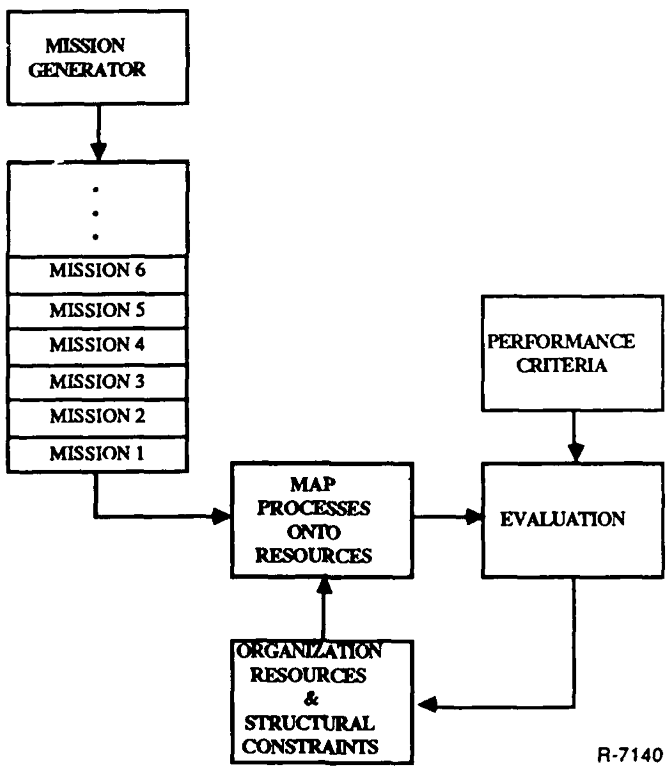



Figure 1-1. Processes Required in the Design of an Organizational Structure.

The present report addresses the problem of optimal mapping of <u>processes</u> onto <u>resources</u>, some of which may be <u>organizational elements</u>, in order to meet specified performance objectives or <u>goals</u> while taking into account various types of constraints

(i.e., organizational, resource, communication, and coordination). Stated another way, the solution to this mapping problem defines the goal and process ATO matrices in Table 1-1. For the system in Fig. 1-1 we are attempting to solve only the mapping problem. While the design of an optimal organization is not considered in this report, the technique does provide a means for evaluating alternative organizations with respect to their ability to meet mission needs.

In order to demonstrate the ability of the optimal mapping technique to address the assignment of processes to resources in a military organization, an example is developed using the Tactical Air Force's Control and Reporting Center (CRC). Finally, a sensitivity analysis technique using Petri nets is illustrated.

## SECTION 2

## MAPPING PROCESSES ONTO $C^3$ ORGANIZATIONS

### 2.1 INTRODUCTION

The goal of a military organization is to accomplish assigned missions successfully and efficiently. To evaluate how successful and efficient an organization is, we decompose the mission into specific processes with the constraint that the processes have to be completed in a certain order (i.e., according to a plan). The organization is also decomposed into individual operating elements or units (humans, groups of humans, etc.) with assigned resources. The communication links between these resources define the hierarchy in the organization; in general, resources have different capabilities and perform some processes better than others.

Assume that a sequence of processes must be completed by a $C^3$ organization. The sequence in which the processes are to be completed can be represented by a finite, acyclic, directed graph.

$$G_p = (V_p, E_p) \text{ with } [V_p] = N \tag{2-1}$$

where $V_p$ is the set of nodes in the graph denoting processes and $E_p$ the set of edges (ordered pairs) denoting data dependencies. Without loss of generality one can assume that there is a starting process $P_1$ and a terminal process $P_t$. Note that the process graph is a generalization of the POF and COF resource matrices that stipulates <u>precedence</u> relationships among the processes. The process graph can also be viewed as a means of defining subgoals (i.e., subprocess objectives) from an overall mission goal.

The organization performing the processes can be represented by an undirected graph

$$G_r = (V_r, E_r) \text{ with } [V_r] = M \tag{2-2}$$

where $V_r$ is the set of nodes in the graph denoting resources and $E_r$ is the set of edges (pairs of nodes) denoting communication links. Note that the resource graph is a symmetric form of the STO and RFM resource matrices.

Figure 2-1 shows an example of a process graph and an example of a resource graph. In this example one would represent the two graphs as follows:

$$G_p = (V_p, E_p) \qquad (2\text{-}3)$$

$$V_p = \{P_1,P_2,P_3,P_4,P_5,P_6,P_7\} \qquad (2\text{-}4)$$

$$E_p = \{(P_1 \rightarrow P_2),(P_1 \rightarrow P_3),(P_2 \rightarrow P_4),(P_2 \rightarrow P_5),(P_3 \rightarrow P_5),(P_3 \rightarrow P_6),(P_4 \rightarrow P_7),(P_5 \rightarrow P_7),(P_6 \rightarrow P_7)\} \qquad (2\text{-}5)$$

$$G_r = (V_r, E_r) \qquad (2\text{-}6)$$

$$V_r = \{R_1,R_2,R_3,R_4\} \qquad (2\text{-}7)$$

$$E_r = \{(R_1,R_2),(R_1,R_3),(R_1,R_4),(R_2,R_3)\} \qquad (2\text{-}8)$$

where $P_1$ is the starting process and $P_7$ is the terminal process. The process graph defines precedence constraints; for example process $P_1$ has to finish first, then $P_2$ and $P_3$ can start and process $P_5$ can proceed after the completion of $P_2$ and $P_3$, etc. With the resource graph one can define the hierarchy and communication paths between resources. For example, resource $R_1$ can communicate directly with all the other resources, but $R_2$ cannot communicate directly with $R_4$.

In addition to the precedence constraints defined by the directed edges in the process graph, each process has the following characteristics (also summarized in Table 2-1):

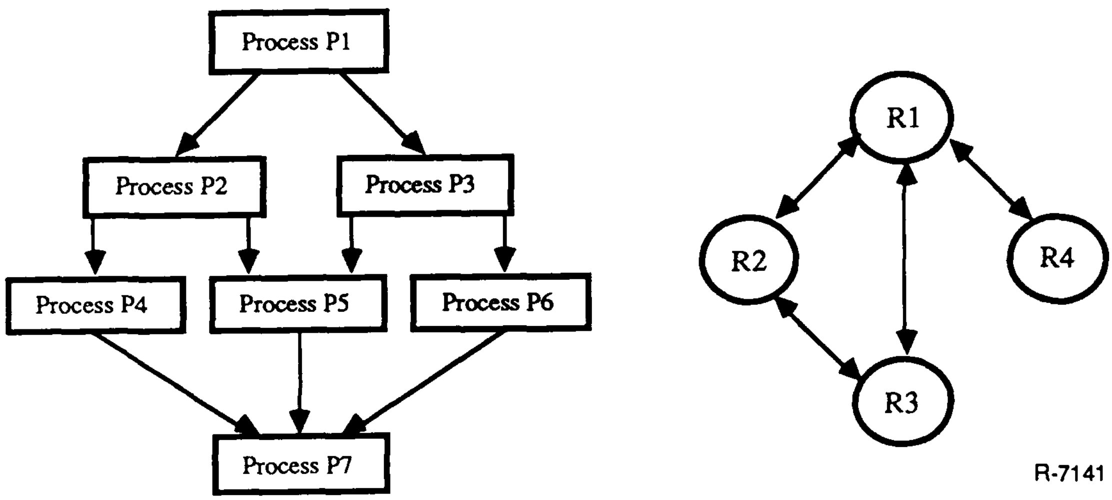



Figure 2-1. Example Process and Resource Graphs.

TABLE 2-1. PROCESS CHARACTERISTICS

| | | |
|---|---|---|
| PC1: | Graph of processes with precedence constraints | $G_p = (V_p, E_p)$ |
| PC2: | Amount of information transmitted from process $P_i$ to $P_j$ | $v_{ij}$ |
| PC3: | Number of resources required to complete process $P_i$ (same starting time) | $n_i$ |
| PC4: | Process difficulty | $d_i$ |
| PC5: | Consistency/Continuity constraint | $g_k$ |

a) Amount of information transmitted between processes. For example, with the process subgraph of Fig. 2-2 process $P_k$ cannot start unless process $P_i$ and process $P_j$ are completed. A certain amount of information has to be transmitted from processes $P_i$ and $P_j$ to enable process $P_k$ to proceed. It is clear that process $P_k$ has to be connected with processes $P_i$ and $P_j$ in the process graph, with directed edges so they can communicate. Depending on the type of exchanged data, one could define a metric to consistently represent the amount of transmitted information.

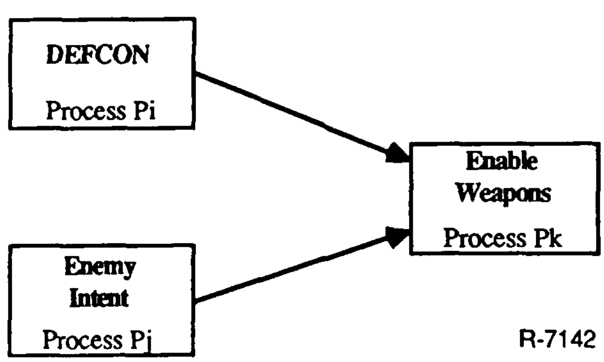



Figure 2-2. An Example of Process Precedence.

b) To complete certain processes you may have to use more than one resource. We assume that these multiple resources will start operating at the same time for each process. For example, let us assume that one of the processes in a mission is to have a conference of three officers to assess a situation. In such a case, more than one resource (officer) is needed to complete the process.

c) Process difficulty is a discrete variable representing the mental or physical requirements of the process. A metric is required to classify the different levels of physical effort, mental effort, and stress induced by the process. As an example, one could classify the processes as "light" with difficulty of .5, "average" with difficulty of 1 and "heavy" with difficulty of 2. With such a metric one can impose the following constraint: A resource can only perform 6 average or 3 heavy or 12 light processes in a given time period. The quantization of process difficulty and the relation of the difficulty of one process to the others depends on the specific application. Even though the relative difficulty of the processes is important, the actual metric used is arbitrary. In the previous example one could have as scales of process difficulty 1, 2, 4 instead of .5, 1, 2 and a resource constraint of 12 average processes instead of 6.

d) Certain processes may be similar in nature and may be required to be performed by the same resource. A consistency/continuity constraint has to be introduced where we define sets of processes $g_k$ with the requirement that a single resource has to perform all the processes in each set. As an example, if we define the following two sets,

$$g_1 = \{P_1, P_2, P_4\} \tag{2-9}$$

$$g_2 = \{P_3, P_5\} \tag{2-10}$$

we indicate that the processes in set $g_1$ have to be completed by a single resource. The processes in set $g_2$ also have to be completed by a single resource (which can be different from the resource performing the processes in $g_1$).

In addition to the superimposed organization structure, the resources have the following characteristics (also summarized in Table 2-2):

a) Time required for resource $R_p$ to complete process $P_i$. One can define a matrix with time entries for all the possible resource-process combinations. If a resource $R_p$ cannot or should not perform the process $P_i$ (i.e., the corresponding element of the resource x process AST matrix is zero), then the time required for the resource $R_p$ to complete process $P_i$ is $\infty$ (or a relatively large number).

b) The communication rate between resources $R_p$ and $R_q$ is measured in message-units/sec. The communication rate ($\mu_{pq}$) can be affected by the distance between resources, the equipment available to them, their expertise and experience, etc. If two resources are not connected by a communication link in the resource graph (i.e., the corresponding elements of the STO and RFM resource matrices are zero), then we define the path between them such that the communication rate between

TABLE 2-2. RESOURCE CHARACTERISTICS

| | | |
|---|---|---|
| OC1: | Graph of resources with precedence constraints | $G_r = (V_r, E_r)$ |
| OC2: | Time required for resource $R_p$ to complete process $P_i$ | $t_{ip}$ |
| OC3: | Communication rate between resources $R_p$ and $R_q$ | $\mu_{pq}$ |
| OC4: | Workload capacity | $L_p$ |

resource $R_p$ and resource $R_q$ is minimum. That is the path $(p\, r_1^* r_2^* \ldots r_n^*\, q)$ is selected such that:

$$\mu_{pq}^{-1} = \min_{r_1 \ldots r_n} \left(\frac{1}{\mu_{p r_1}} + \ldots + \frac{1}{\mu_{r_n q}}\right) = \left(\frac{1}{\mu_{p r_1^*}} + \ldots + \frac{1}{\mu_{r_n^* q}}\right) \tag{2-11}$$

where $(pr_1 \ldots r_nq)$ are all the paths connecting resource $R_p$ and resource $R_q$.

For example in the resource graph in Fig. 2-1, resources $R_2$ and $R_4$ are not connected directly so the minimum existing path and the communication rate $\mu_{24}$ are defined by

$$\mu_{24}^{-1} = \min_{\substack{(2,1,4),\\ (2,3,1,4)}} \left[\left(\frac{1}{\mu_{2,1}} + \frac{1}{\mu_{1,4}}\right), \left(\frac{1}{\mu_{2,3}} + \frac{1}{\mu_{3,1}} + \frac{1}{\mu_{1,4}}\right)\right] \tag{2-12}$$

where the path that leads to the minimum communication rate $\mu_{24}$ is either (2, 1, 4) or (2, 3, 1, 4) depending the communication rates $\mu_{2,1}$, $\mu_{1,4}$, $\mu_{2,3}$, and $\mu_{3,1}$.

In the event that no connecting paths exist between two resources, the communication rate between those resources is 0.

c) The limit on the processes performed by a resource is a constraint on the total amount of work that the resource is capable of performing. This is an indirect way to model the <u>workload capacity</u> of the resource. The limit should be defined in accordance with the process difficulty metric defined previously.

After a mission is translated to a sequence of processes, the problem that the $C^3$ analyst faces is to allocate these processes to the resources so that the completion time of the final process is minimized and at the same time none of the constraints is violated. In the sequel we will formulate the problem in detail.

## 2.2 FORMULATION OF THE MAPPING PROBLEM

Assume that resource $R_p$ is assigned to perform a certain sequence of processes. This sequence will be denoted as a set of ordered pairs

$$Q_p = \{(P_n, s_{np}), \ldots, (P_m, s_{mp})\} \qquad (2\text{-}13)$$

where $P_n,..,P_m$ are the processes to be performed by $R_p$ and $s_{np},...,s_{mp}$ are the starting times of the corresponding processes. That is to say, $R_p$ will start performing process $P_n$ at time $s_{np}$, etc. For example in the time diagram of Fig. 2-3, resource $R_2$ will start performing process $P_2$ at time $s_{2,2}$, and then at time $s_{4,2}$ process $P_4$ will begin, etc.

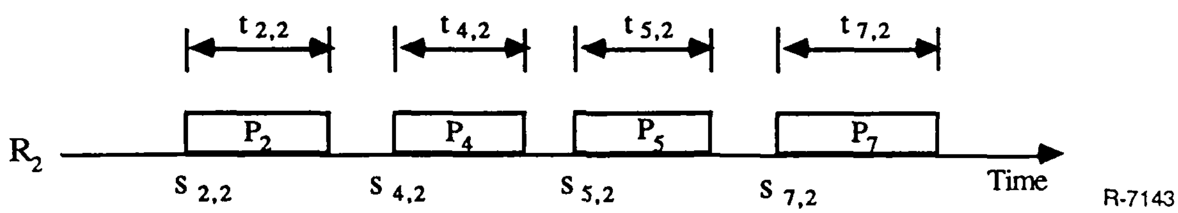



Figure 2-3. An Example of a Timing Diagram for Resource $R_2$.

The objective is to find the order of process completion for each resource ($Q_p$, $\forall p$) such that the completion time $C_t = s_{tp} + t_{tp}$ of the terminal process $P_t$ is minimized

$$\min_{R_p \in V_p} (s_{tp} + t_{tp}) \qquad (2\text{-}14)$$

subject to the following constraints:

a) Each resource $R_p$ requires $t_{np}$ amount of time to complete process $P_n$ and each resource can perform only one process at a time.

b) If resource $R_p$ is performing process $P_i$ and resource $R_q$ is performing process $P_j$ and process $P_i$ precedes process $P_j$ in the process graph, the following must hold:

$$s_{jq} \geq s_{ip} + t_{ip} + \frac{v_{ij}}{\mu_{pq}} \qquad (2\text{-}15)$$

In addition, each communication link can transmit only one message at a time.

c) The processes allocated to a resource should not violate the workload capacity of the resource. More specifically the following must hold:

$$\sum_{(P_i, s_{ip}) \in Q_p} d_i \leq L_p \quad \forall\ p=1,2,\ldots,M \qquad (2\text{-}16)$$

d) If many resources ($n_i$) are required to complete a process, we assume they must overlap at some time. The current version of the algorithm assumes that all the resources have to <u>start</u> that process at the same time. Specifically:

$$\forall i,p,q \quad \text{if } (P_i,s_{ip}) \in Q_p \text{ and } (P_i,s_{iq}) \in Q_q \qquad \text{then} \quad s_{i\,p} = s_{iq} \qquad (2\text{-}17)$$

and process $P_i$ will be processed by exactly $n_i$ resources. (Note: This constraint was specifically incorporated to represent group decisionmaking, conferences, etc. There are a variety of other ways in which similar notions may be captured in future modifications to Mapper.)

e) All processes that are similar in nature have to be completed by the same resource. As stated previously, each set $g_k$ consists of processes that are similar. The resource allocation should be such that for all the processes in $g_k$ there exists a resource $R_p$ that can execute them. To be more specific:

$$\forall g_k\ \exists p \text{ such that } \forall P_i \in g_k\ (P_i,s_{ip}) \in Q_p \qquad (2\text{-}18)$$

It is also assumed that a resource can perform a process and communicate with another processor simultaneously (in parallel). If the resources happen to be airplanes, ships, computers, etc., this may be a realistic assumption, but for human resources this may not be the case. In future work we should relax this assumption so that some resources can do the communication and processing only sequentially.

## 2.3 SOLUTION OF THE MAPPING PROBLEM

### 2.3.1. Computational Complexity

To illustrate the problem complexity, consider the case when there are no constraints on workload capacity or consistency/continuity, with the same starting time for processes that have to be performed by multiple resources. Then the total number of different allocations of M resources for a process is $\prod_{a=0}^{n_i-1} (M - a)$. The execution order of the N processes (i.e., the number of possible execution sequences of processes) is upper bounded by N!. Therefore, the

total number of different allocations in the worst case is $N! \prod_{i=1}^{N} \prod_{a=0}^{n_i - 1} (M - a)$. When $n_i = 1$ for $0 \le i \le N$, the total number of different allocations in the worst case is $N!\, M^N$.

Indeed, the resource allocation problem based above is NP-hard [5]-[8], which means that our optimal algorithm for the resource allocation problem with a run-time bound that is a polynomial function of the number of resources exists if, and only if, a class of combinatorial optimization problems, including the traveling salesman, maximum clique, and the satisfiability problems can be solved in polynomial time [8], [9]. The evidence indicates that in all likelihood any problem which is NP-hard cannot be solved by an algorithm of polynomial time complexity. Therefore, all practical algorithms exploit the use of heuristics to reduce the computational requirements. In the following, we develop a heuristic mapping algorithm, which has been proven to be asymptotically optimal for series-parallel process structures in [5]. For ease of exposition, we first develop the allocation algorithm for the case where there are no continuity/consistency or the same starting time constraints, and then extend to the general case.

### 2.3.2 Key Allocation Equation

The workload capacity constraints simply restrict the feasible resource assignments for a process. The constraint that a process i requires $n_i$ ($\ge 1$) resources, however, imposes additional synchronization delays since a process $P_i$ cannot begin execution until the information from all the resources that complete the parents (predecessors) of process $P_i$ is available at the assigned resource. In addition, the ordered sets $Q_p$ are not disjoint, since a process $P_i$ must be acted on by multiple resources.

To drive the allocation algorithm, let $(Q_1, Q_2, ..., Q_M)$ denote partial allocation when process $P_i$ is being considered for allocation. In addition, let $F_i$ denote the set of feasible resource assignments to which a ready process $P_i$ can be assigned without violating the workload capacity constraints, i.e.,

$$F_i = \left\{ R_q : d_i + \sum_{(P_j,\, s_{jq}) \in Q_q} d_j \leq L_q \right\} \tag{2-19}$$

The completion time of process $P_i$ at resource $R_q \in F_i$ is a function of:

1. the service time of process $P_i$ at resource $R_q$, $t_{iq}$ ;
2. the time at which the data from immediate predecessors of process $P_i$ is available at resource $R_q$; and
3. the time when resource $R_q$ becomes available (i.e., the completion time of the last process, say $P_a$, executed on resource $R_q$, $C_a$).

Formally,

$$C_i\,(Q_1, Q_2, \ldots, Q_q \cup \{P_i, s_{iq}\}, \ldots, Q_M) = s_{iq} + t_{iq} \tag{2-20}$$

where $s_{iq}$ is the starting time of process $P_i$ on resource $R_q$, and $t_{iq}$ is the corresponding service time. As discussed previously, the starting time, in turn, is a function of when the information from the resource set executing the parent processes of process $P_i$ becomes available at resource $R_q$, and the time at which resource $R_q$ is available. Specifically,

$$s_{iq} = \max\,(D_{iq}, C_a\,(Q_1, Q_2, \ldots, Q_q, \ldots Q_M)) \tag{2-21}$$

where $D_{iq}$ is the time at which data from all parents of process $P_i$ become available at resource $R_q$, and $P_a$ (with completion time $C_a$) is the last process in the ordered set $Q_q$. The earliest time at which data from all the parents of process $P_i$ become available at resource $R_q$ is:

$$D_{i\,q} = \max_{P_j \in \beta_i} \left\{ C_j(Q_1, Q_2, \ldots, Q_M) + \frac{1}{\mu_{q'_j q}} \right\} \tag{2-22}$$

where $\beta_i$ is the set of predecessor processes of process $P_i$, and $R_{q'_j}$ is the resource to which process $P_j$ is assigned. The computation implied by Eqs. 2-21 and 2-22 is illustrated in Fig. 2-4.

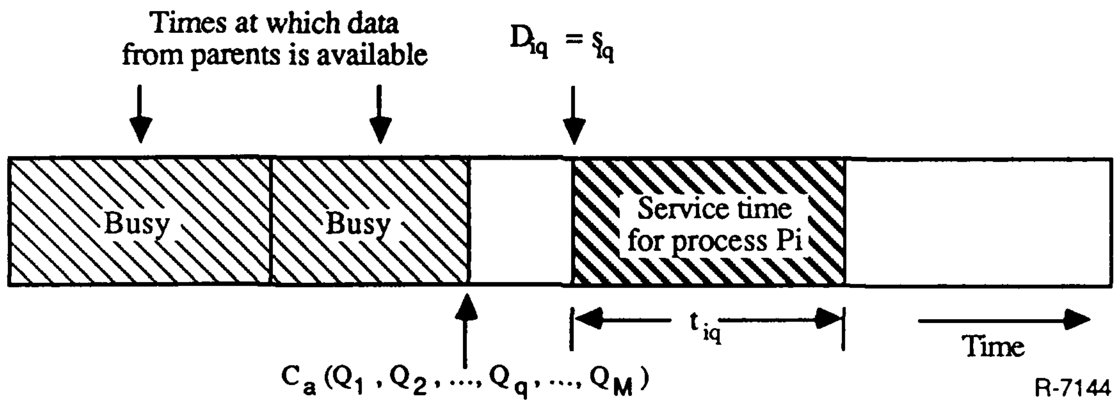



Figure 2-4. Example Timing Diagram for Resource $R_q$.

### 2.3.3. The Heuristic Algorithm

The heuristic algorithm consists of two stages. The first stage employs the concept of critical path to determine the order of process execution, while the second stage sequentially allocates the tasks from the ordered list to resources so that the completion time of the processes is a minimum. To determine the order of process allocation, we define the level of process $P_i$ as

$$A_i = \max_k \sum_{j \in \Pi_k^i} \min_{q \in V_r} t_{jq} \tag{2-23}$$

where $\Pi_k^i$ denotes the $k^{th}$ path from process $P_i$ to the terminal process $P_t$. By construction, the optimal completion time of the terminal process $P_t$, $C_t^*$, is an upper bound on the level of <u>any</u> process $P_i$. That is,

$$C_t^* \geq \max_i A_i \tag{2-24}$$

Following the level algorithm of [10] and Sethi [11], the heuristic algorithm is based on the premise that processes with larger levels should be executed earlier in the sequence. If several processes $P_i$ have the same level, then the process with the greater number of successors should be executed first. Thus, we construct the execution order of processes according to nonincreasing levels first, and nonincreasing successors next if processes have the same level.

Once the priority list of process execution is constructed, we sequentially allocate processes to resources to minimize the completion time. Specifically, we assign process $P_i$ of the priority list to $n_i$ distinct resources $R^*_{q_1}, R^*_{q_2}, \ldots, R^*_{q_{n_i}}$ that yield minimum completion time. That is, the assignments $R^*_{q_k}$ $(1 \le k \le n_i)$ are such that:

$$\begin{aligned}
&C_i\left(Q_1, Q_2, \ldots, Q^*_{q1} \cup \left\{P_i, s^*_{i\,q_1}\right\}, \ldots, Q_M\right) \\
&\le C_i\left(Q_1, Q_2, \ldots, Q^*_{q2} \cup \left\{P_i, s^*_{i\,q_2}\right\}, \ldots, Q_M\right) \le \ldots \\
&\le C_i\left(Q_1, Q_2, \ldots, Q^*_{qni} \cup \left\{P_i, s^*_{i\,q_{n_i}}\right\}, \ldots, Q_M\right) \le \ldots \qquad (2\text{-}25)\\
&\le C_i\left(Q_1, Q_2, \ldots, Q^*_{q\,|F_i|} \cup \left\{P_i, s^*_{i\,q\,|F_i|}\right\}, \ldots, Q_M\right)
\end{aligned}$$

where $F_i$ was defined earlier in Eq. 2-19. The complexity of the heuristic mapping algorithm is O(MN). The heuristic algorithm proceeds as follows:

Given a directed process graph $G_p = (V_p, E_p)$ and resource graph $G_r = (V_r, E_r)$ with parameters $(v_{ij}, n_i, d_i, t_{ip}, L_p, \mu_{pq})$, Mapper computes the allocations $(Q_1, Q_2, \ldots, Q_M)$. Let $\gamma_i$ denote the set of immediate successors (children) of process $P_i$ and $\beta_i$ represent the set of immediate predecessors (parents) of process $P_i$.

***Step 1:*** Determine the level of each process

$Z = \{P_t\}$

Repeat until Z is empty

Select a process $P_i$ of Z such that no successors of process $P_i$ appear in Z

Compute the level of processor $P_i$ via

$$A_i = \max_{j \in \gamma_i} \{A_j\} + \min_{q \in V_p} t_{iq}$$

$$Z = Z - \{P_i\} \cup \beta_i$$

end

**Step 2:** Construct a priority list [1] [2] ... [N] by sorting $A_i$ in nonincreasing order. Break ties on the basis of number of successors, with the more successors given the higher priority.

**Step 3:** $Q_q = 0 \ \forall_q$

For j = 1 to N DO

$i = [j]$

Form a feasible set of processors $F_i$ via Eq. 2-19

Find assignments $\overset{*}{q_k}$ $(1 \leq k \leq n_i)$ via Eq. 2-25

$Q_{q_k}^* = Q_{q_k}^* \cup \left\{P_i, s_{iq_k}^*\right\}$

end

### 2.3.4. Extension to Include Consistency/Continuity and Same-Starting-Time Constraints

Suppose a process $P_i$ requires $n_i(>1)$ resources for execution. Further, it is required that all $n_i$ resources start execution on process $P_i$ at <u>the same time</u>. This is easily accomplished in Mapper by changing the starting times of resources $R_{q_k}^*$ $1 \leq k \leq n_i$ to the latest starting time in Eq. 2-25. That is,

$$s_{i\overset{*}{q_k}} = \max_k \left[ C_i\left(Q_1, Q_2, \ldots, Q^*_{q_k} \cup \left\{P_i, s_{i\overset{*}{q_k}}\right\}, Q_M\right) - t_{i\overset{*}{q_k}} \right]; 1 \leq k \leq n_i \qquad (2\text{-}26)$$

The consistency/continuity constraint requires that groups of processes $g_k$ should be allocated to a single resource, if the completion time does not exceed a prespecified bound; that is, the group of processes $g_k$ should be allocated to a single resource if $C_t^1 \leq (1 + \varepsilon)\, C_t$ where $C_t^1$ is the completion time when $g_k$ is allocated to a single resource, $C_t$ is the completion time when the consistency constraint is removed, and $\varepsilon$ is a user-specified acceptable bound (e.g., $\varepsilon = 0.1$ for a 10 percent deviation). It is assumed that groups of process $g_k$, k=1,2,..., NG are prioritized with NG being the total number of groups (sets of processes); the group $g_1$ has the highest priority and the set $g_{NG}$ has the lowest priorty. Mapper extends the algorithm of subsection 2.3.3 to include the consistency/continuity constraints as follows:

Given a directed processor graph $G_p = (V_p, E_p)$ and resource graph $G_r = (V_r, E_r)$ with the parameters ($V_{ij}$, $n_i$, $d_i$, $t_{ip}$, $L_p$, $\mu_{pq}$), and consistency groups $g_k$, $(1 \le k \le NG)$, Mapper computes allocations ($Q_1, Q_2, \ldots, Q_M$), querying the user regarding the acceptability of the time penalty for each consistency group in order.

**Step 1:** Apply heuristic allocation algorithm without consistency/continuity constraints. Let $C_{th}$ be the completion time of the terminal task. Let [1] [2] ... [N] be the priority list.

Initialize selection flags $IG(k) = 0$

**Step 2:**

```
For all groups g_k, 1 ≤ k ≤ NG, DO
    For j = 1,2,..., N  DO
        i = [j]
        If i ∈ g_m' where m' ∈ (1,2,..., k-1) and IG(m') ≠ 0, update
            resource available times and completion time of P_i
        else
            if i ∈ g_k
                if i = i_kf, the first process in g_k,
                    Find assignments of process P_i as in
                        Eq. 2-25
                    Assign other process in g_k to the same set
                        of resources
                else
                    if i ≠ i_kf and n_i > n_{i_kf}
                        Find assignments of process P_i
                            for n_{i_kf} + 1, ..., n_i as in
                            Eq. 2-25
                end if
            else
                Find assignments of process P_i as in Eq. 2-25
            end if
        end if
    end
    If C_t ≤ (1 + ε) C_th then
        IG(k) = 1
        Find assignments for group g_k
    end if
end
```

## 2.4 EXAMPLE

To illustrate the potential applicability of the algorithm described in subsection 2.3 we generate a realistic mission for a Control and Reporting Center (CRC) (see [12]) and we solve the process mapping problem. More specifically, consider a simplified (and hypothetical) CRC organization, given in Fig. 2-5, with the following resources:

a) Battle Commander ("COMMANDER" in the figure), the senior Air Force Officer at the CRC. Minimum qualifications will be a field grade officer possessing a current 1716/1744 Air Force Specialty Code (AFSC).

b) Senior Director ("SENIOR DIRECTOR") (with 1716/1744 AFSC).

c) Weapon Assignment Officer ("WAO") (with 1744 AFSC).

d) Air Surveillance Officer ("SURVEILLANCE") (with 1744 AFSC).

e) Weapons Controller ("WC") (with 1744 AFSC).

f) Movements and Identification Supervisor ("M&I") (with 27670 AFSC).

g) Staff member ("STAFF") (with 27650 AFSC).

h) Staff member ("STAFF") (with 27650 AFSC).

i) Staff member ("STAFF") (with 27650 AFSC).

The organizational structure shown in Fig. 2-5 was constructed using reference [12] for the purpose of illustrating the process mapping problem and it does not represent any existing organization.

A hypothetical mission is postulated with 37 processes. The mission's objective is to track, identify, and prosecute 10 aircraft that penetrated the airspace under the control of the CRC. The described mission will be used to illustrate the process mapping problem and it is not intended to describe an actual mission. Fig. 2-6 shows the process graph for this mission, and the processes are defined as follows:

a) Track Initiation for 10 targets ($P_1$). This is the starting process where 10 potential targets were identified by surveillance and their tracks were initiated.

b) Movement and Identification operation for each target ($P_2$, $P_3$, $P_4$, $P_5$, $P_6$, $P_9$, $P_{10}$, $P_{11}$, $P_{12}$, $P_{13}$). These processes are responsible to insure that proper

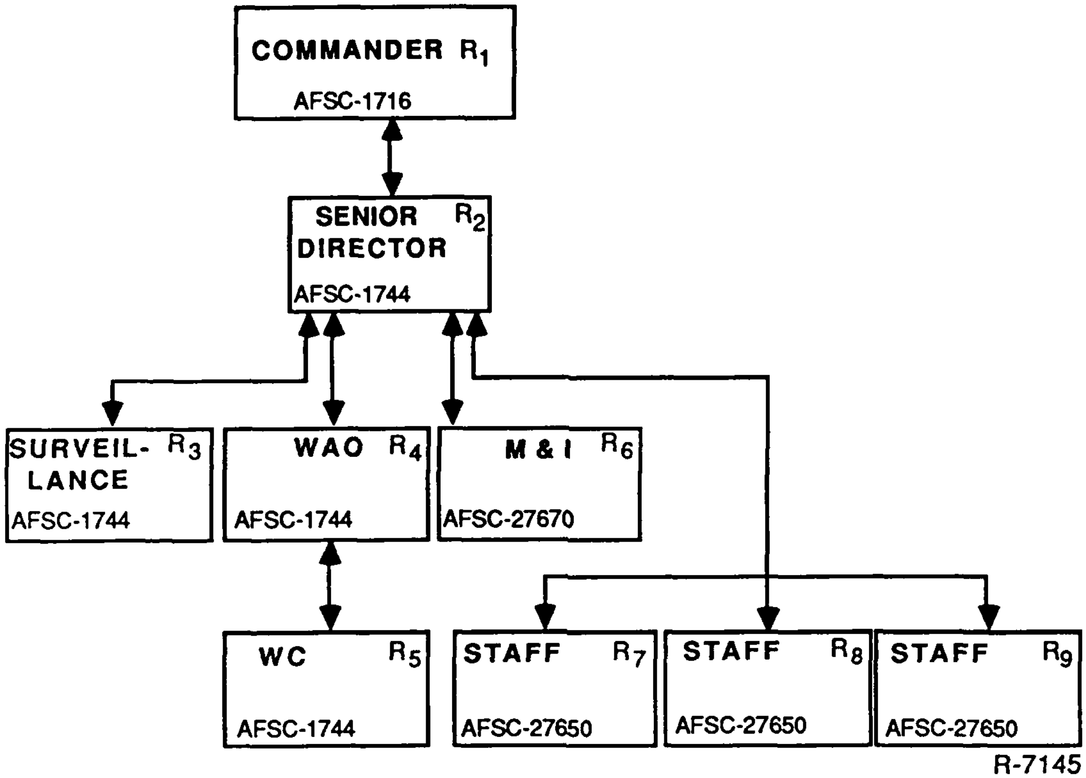



Figure 2-5. Resource Organization

classification has been assigned to all tracks. In fact, in this mission it is assumed that processes $P_2$, $P_3$, $P_4$, $P_5$, and $P_6$ are checking all ten targets and that they identified only 5 of them. Then processes $P_9$, $P_{10}$, $P_{11}$, $P_{12}$, and $P_{13}$ continued the identification for the remaining 5 targets.

c) Supervision of the Movement and Identification operation ($P_7$, $P_{14}$). This process is responsible for supervising the movement and identification operations. Specifically, process $P_7$ is supervising and providing direction to processes $P_2$, $P_3$, $P_4$, $P_5$, $P_6$ and process $P_{14}$ is supervising processes $P_9$, $P_{10}$, $P_{11}$, $P_{12}$, and $P_{13}$.

d) Situation Assessment ($P_8$, $P_{17}$). Process $P_8$ is responsible for assessing the situation after the movement and identification of the 10 targets where only 5 were identified, and process $P_{17}$ is assessing the situation after the remaining 5 targets were identified.

e) Planning ($P_{15}$, $P_{19}$). Processes $P_{15}$ and $P_{19}$ are generating the plan, coordinating the activities with other units, and defining the activities that have to take place in the CRC. Usually the planning processes require more than one resource.

f) Information update and exchange ($P_{16}$, $P_{20}$, $P_{29}$, $P_{34}$). Update and check the accuracy of the information on the vertical plotting board.

g) Aircraft Allocation ($P_{18}$, $P_{22}$). Responsible for assigning aircraft to the mission.

h) Weapon and Target Pairing ($P_{21}$, $P_{24}$).

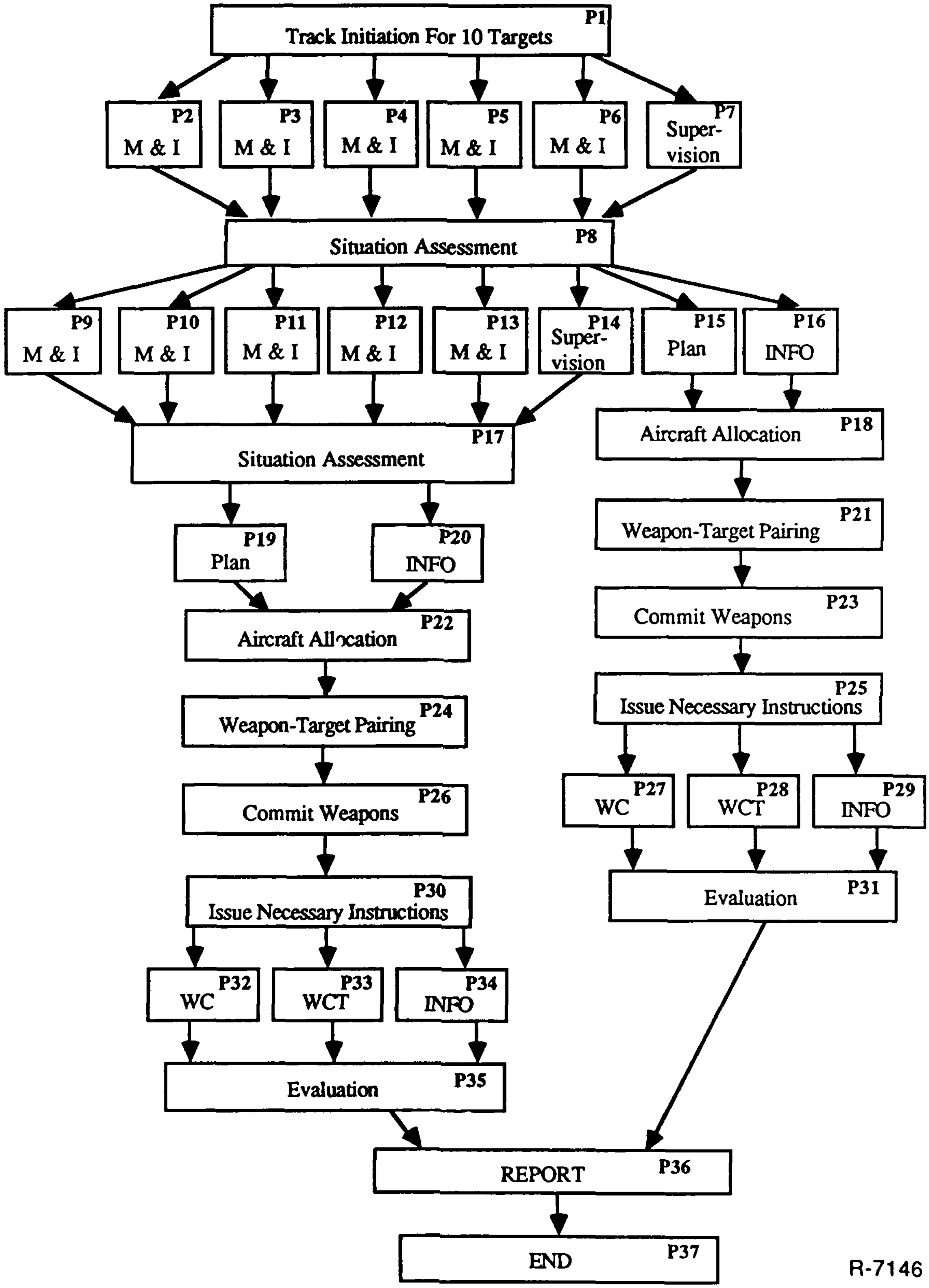



Figure 2-6. Process Precedence Constraints.

i) Commit Weapons ($P_{23}$, $P_{26}$). Authorize the weapon/target pairing and the aircraft assignments.

k) Issue Necessary Instructions ($P_{25}$, $P_{30}$). Coordinate activities by issuing the necessary instructions.

l) Weapon controlling activity ($P_{27}$, $P_{32}$). Issue such instructions as necessary to insure an orderly flow of aircraft to assigned targets, obtain pilot weather reports and other information, provide any assistance needed to aircrews.

m) Weapon controlling support ($P_{28}$, $P_{33}$). Insure that adequate supplies and support are available so that the weapon controlling activity is successful.

n) Evaluation of battle ($P_{31}$, $P_{35}$)

o) Reporting ($P_{36}$)

p) End of mission ($P_{37}$).

Tables 2-3 through 2-5 identify some of the characteristics of the processes for this mission and the characteristics of the resources in the CRC organization. As described in subsection 2.2, these characteristics are used to obtain the constraints for the mapping problem. Specifically, the processes and resources for this example have the following characteristics:

PC1: The graph of processes $G_p = (V_p, E_p)$ with precedence constraints is given in Fig. 2-6.

PC2: The amount of information ($v_{ij}$) transmitted from process $P_i$ to $P_j$ is 1 message for all i and j.

PC3: The number of resources ($n_i$) required to complete process $P_i$ is given in Table 2-3.

PC4: The process difficulty ($d_i$) is assumed to be the same for all processes and, therefore, $d_i=1\ \forall i$.

PC5: There is no consistency/continuity requirement for the first case. A consistency/ continuity requirement will be added for the next case.

OC1: The graph of resources $G_r = (V_r, E_r)$ with precedence constraints is given in Fig. 2-5.

OC2: The time ($t_{ip}$) required for each resource $R_p$ to complete each process $P_i$ is given in Table 2-4.

OC3: The communication rates ($\mu_{pq}$) between resources for each resource $R_p$ and resource $R_q$ are given in Table 2-5.

OC4: The work load capacity ($L_p$) for resources is given in Table 2-5.

## TABLE 2-3. PROCESS CONSTRAINTS

| Process # | Process name | Resources required to complete process | Process difficulty |
|---|---|---|---|
| $P_1$ | Track initiation for 10 targets | 1 | 1 |
| $P_2$ $P_3$ $P_4$ $P_5$ $P_6$ $P_9$ $P_{10}$ $P_{11}$ $P_{12}$ $P_{13}$ | M&I | 1 | 1 |
| $P_7$ $P_{14}$ | Supervision | 1 | 1 |
| $P_{15}P_{19}$ | Plan | 3 | 1 |
| $P_{16}P_{20}P_{29}P_{34}$ | Information Exchange | 1 | 1 |
| $P_8$ $P_{17}$ | Situation Assessment | 2 | 1 |
| $P_{18}P_{22}$ | Aircraft Allocation | 1 | 1 |
| $P_{21}P_{24}$ | Weapon/Target Pairing | 2 | 1 |
| $P_{23}P_{26}$ | Commit Weapons | 2 | 1 |
| $P_{25}P_{30}$ | Issue Necessary Instructions | 2 | 1 |
| $P_{27}P_{32}$ | WC | 2 | 1 |
| $P_{28}P_{33}$ | WCT | 3 | 1 |
| $P_{31}P_{35}$ | Evaluation | 2 | 1 |
| $P_{36}$ | Report | 2 | 1 |
| $P_{37}$ | end | 1 | 1 |

R-7192

TABLE 2-4. TIME REQUIRED BY RESOURCES TO COMPLETE PROCES[illegible] N)

| Process # | Process name | (Commander) $R_1$ | (Senior Director) $R_2$ | (Surveillance) $R_3$ | (WAO) $R_4$ | (WC) $R_5$ | (M&I) $R_6$ | (Staff) $R_7$ | (Staff) $R_8$ | (Staff) $R_9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $P_1$ | Track initiation for 10 targets | ∞ | ∞ | 10 | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ |
| $P_2$ $P_3$ $P_4$ $P_5$ $P_6$ $P_9$ $P_{10}$ $P_{11}$ $P_{12}$ $P_{13}$ | M&I | ∞ | ∞ | ∞ | ∞ | ∞ | 10 | 20 | 20 | 20 |
| $T_7$ $T_{14}$ | Supervision | 20 | 10 | ∞ | ∞ | ∞ | 5 | ∞ | ∞ | ∞ |
| $P_{15}$ $P_{19}$ | Plan | 5 | 5 | ∞ | 10 | ∞ | 20 | ∞ | ∞ | ∞ |
| $P_{16}$ $P_{20}$ $P_{29}$ $P_{34}$ | Information Exchange | ∞ | ∞ | ∞ | ∞ | 10 | 10 | 5 | 5 | 5 |
| $P_8$ $P_{17}$ | Situation Assessment | 10 | 10 | 5 | 5 | 20 | 5 | ∞ | ∞ | ∞ |
| $P_{18}$ $P_{22}$ | Aircraft Allocation | 10 | 5 | ∞ | 10 | 15 | ∞ | ∞ | ∞ | ∞ |
| $P_{21}$ $P_{24}$ | Weapon/Target Pairing | 20 | 10 | ∞ | 10 | 15 | ∞ | ∞ | ∞ | ∞ |
| $P_{23}$ $P_{26}$ | Commit Weapons | 5 | 10 | ∞ | 15 | 20 | ∞ | ∞ | ∞ | ∞ |
| $P_{25}$ $P_{30}$ | Issue Necessary Instructions | 5 | 5 | ∞ | 10 | ∞ | ∞ | ∞ | ∞ | ∞ |
| $P_{27}$ $P_{32}$ | WC | ∞ | 10 | ∞ | 5 | 5 | ∞ | 20 | 20 | 20 |
| $P_{28}$ $P_{33}$ | WCT | ∞ | 20 | ∞ | 15 | 10 | ∞ | 10 | 10 | 10 |
| $P_{31}$ $P_{35}$ | Evaluation | 5 | 5 | ∞ | 10 | 10 | ∞ | ∞ | ∞ | ∞ |
| $P_{36}$ | Report | ∞ | 30 | ∞ | 15 | 15 | ∞ | 60 | 60 | 60 |
| $P_{37}$ | end | 5 | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ |

R-7193

TABLE 2-5. RESOURCE CONSTRAINTS

Work Load Capacity

| | | |
|---|---|---|
| (Commander) | $R_1$ | 10 |
| (Senior Director) | $R_2$ | 15 |
| (Surveillance) | $R_3$ | 15 |
| (WAO) | $R_4$ | 15 |
| (WC) | $R_5$ | 15 |
| (M&I) | $R_6$ | 20 |
| (Staff) | $R_7$ | 20 |
| (Staff) | $R_8$ | 20 |
| (Staff) | $R_9$ | 20 |

Communication Rate

| | |
|---|---|
| Link R1-R2 | 1 |
| Link R2-R3 | 1 |
| Link R2-R4 | 1 |
| Link R2-R6 | 1 |
| Link R2-R7 | 1 |
| Link R2-R8 | 1 |
| Link R2-R9 | 1 |
| Link R4-R5 | 1 |
| Link R4-R6 | 1 |

R-7194

Using the resource allocation algorithm described in subsection 2.3, we assigned the processes of the mission to the resources of the CRC (as shown in Fig. 2-7). Note that the tasks that had to be completed by more than one resource are starting at the same time. In addition, note that the movement and identification processes ($P_2$-$P_6$ and $P_9$-$P_{13}$) are completed by the Movement and Identification supervisor (Resource $R_6$) and staff members while the supervision and planning ($P_7$, $P_{14}$, $P_{15}$, $P_{19}$) were done by higher ranked officers ($R_1$, $R_2$, $R_4$). Resource $R_1$ completes the final process $P_{10}$ at 228 min. as shown in Table 2-6. The resource utilization for this process mapping is given in Table 2-6.

As an indication of how such parallel processing is decreasing the completion time of the mission, let's assume that there is a "super expert" resource with no workload limitation that can perform each process $P_i$ in $t_i$ time ( $t_i = \min_{P} \{t_{ip}\}$ ). Then we will consider the time required for the centralized solution (the "super expert" resource completes all the processes) and compares it with the completion time of the parallel solution. For this example, the centralized solution has a completion time of 633.2 min. The parallel solution has a completion time of 228 min. Therefore, we define the "speedup" as the ratio of the completion time of the centralized solution and the parallel solution (2.78 in this case).

Even though processes $P_{16}$, $P_{20}$, $P_{29}$, and $P_{34}$ are similar, they were allocated to three different resources ($R_7$, $R_8$, $R_9$). This is happening because the three resources have similar

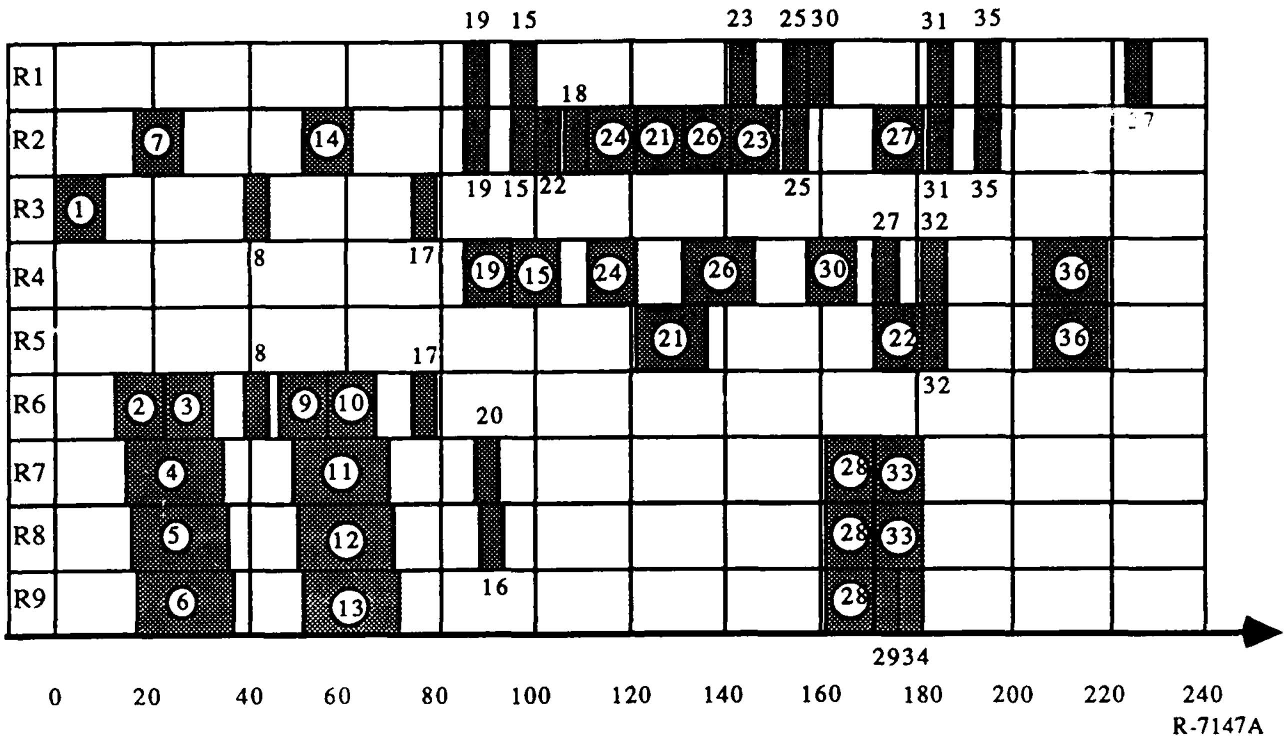



Figure 2-7. Resource Allocation for the CRC Example.

TABLE 2-6. RESOURCE UTILIZATION

| RESOURCE | % UTILIZATION | TOTAL TIME UTILIZED | FINISH TIME |
|---|---|---|---|
| Resource R1 | 17.5 | 40 | 228 |
| Resource R2 | 46.0 | 105 | 197 |
| Resource R3 | 8.8 | 20 | 79 |
| Resource R4 | 35.1 | 80 | 219 |
| Resource R5 | 19.7 | 45 | 219 |
| Resource R6 | 21.9 | 50 | 79 |
| Resource R7 | 28.5 | 65 | 181 |
| Resource R8 | 28.5 | 65 | 181 |
| Resource R9 | 28.5 | 60 | 181 |

characteristics and it does not matter to what resource each of the four processes is allocated. This can be avoided by introducing the consistency/continuity constraint of having the four processes ($P_{16}$, $P_{20}$, $P_{29}$, $P_{34}$) completed by the same resource.

Figure 2-8 shows the resource allocation when the consistency/continuity constraint described above is included in the problem formulation. Note that in this case the four processes ($P_{16}$, $P_{20}$, $P_{29}$, $P_{34}$) were completed by resource $R_7$. The price that we paid for having

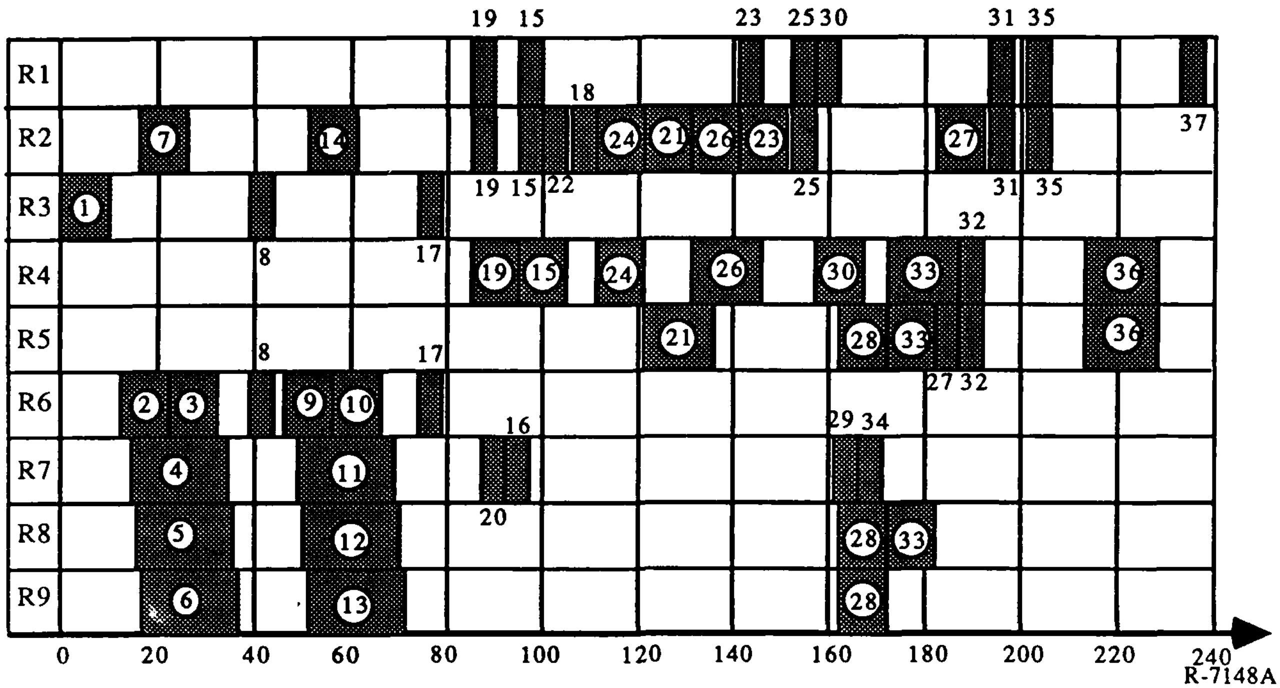



Figure 2-8. Resource Allocation for the CRC Example with the Consistency/ Continuity Constraint.

the four processes completed by the same resource was an approximate 3.9% increase in the completion time. The mission is finished in 237 min. (compared with 228 min. for the previous case); the resource utilization for this process mapping is given in Table 2-7. In this case the centralized solution ("super expert" resource performing the mission) would again take 633.2 min. to complete the mission and the speedup is 2.67.

TABLE 2-7. RESOURCE UTILIZATION

| RESOURCE | %UTILIZATION | TOTAL TIME UTILIZED | FINISH TIME |
|---|---|---|---|
| Resource R1 | 16.8 | 40 | 237 |
| Resource R2 | 44.3 | 105 | 206 |
| Resource R3 | 8.4 | 20 | 79 |
| Resource R4 | 37.9 | 90 | 228 |
| Resource R5 | 25.3 | 60 | 228 |
| Resource R6 | 21.1 | 50 | 79 |
| Resource R7 | 25.3 | 60 | 181 |
| Resource R8 | 25.3 | 60 | 182 |
| Resource R9 | 21.1 | 50 | 172 |

To show the effect of the amount of information transmitted between resources on the completion time of the mission, the same example was used with each process transmitting five messages to other processes instead of one message as in the previous case. Figure 2-9 shows the resulting resource allocation. It is clear that the organization is now behaving in a more centralized manner, with most of the processes completed by $R_2$, $R_4$, and $R_6$. For example, most of the movement and identification activity ($P_2$, $P_3$, $P_4$, $P_6$, $P_9$, $P_{11}$, $P_{12}$, $P_{13}$) is done by the M&I officer and not by the staff as happened in the first two cases (Figs. 2-7 and 2-8). As a result the mission is finished in 490 min. (compared with 228 min. and 237 of the previous cases); the resource utilization for this process mapping is given in Table 2-8. For this last case the speedup is 1.3, which is an indication that the solution for this last case is more centralized than the solution for the previous cases.

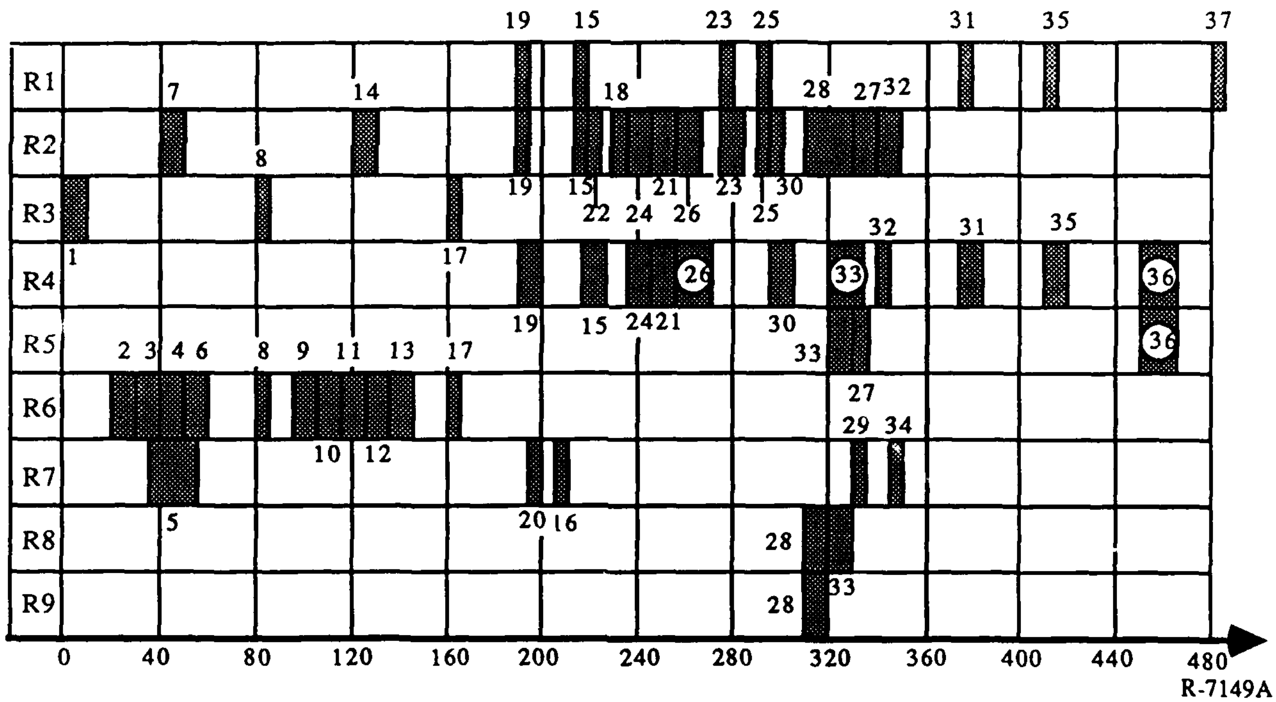



Figure 2-9. Resource Allocation for the CRC Example with Increased Communication Requirements.

TABLE 2-8. RESOURCE UTILIZATION

| RESOURCE | %UTILIZATION | TOTAL TIME UTILIZED | FINISH TIME |
|---|---|---|---|
| Resource R1 | 7.1 | 35 | 490 |
| Resource R2 | 26.5 | 130 | 350 |
| Resource R3 | 4.1 | 20 | 165 |
| Resource R4 | 24.5 | 120 | 465 |
| Resource R5 | 6.1 | 30 | 465 |
| Resource R6 | 20.4 | 100 | 165 |
| Resource R7 | 8.2 | 40 | 350 |
| Resource R8 | 4.1 | 20 | 330 |
| Resource R9 | 2.0 | 10 | 320 |

## SECTION 3

## AN APPROACH TO SENSITIVITY ANALYSIS

### 3.1 SIMULATION OF $C^3$ ORGANIZATIONS USING PETRI NETS

In this section we present a Petri net based methodology for modeling a $C^3$ organization performing a mission. First we introduce two generic models of a resource $R_p$ operating on two processes $P_i$ and $P'_i$, and then we introduce two generic models for using communication links between resources. Finally, we illustrate the use of the Petri net model to determine the sensitivity of the performance of the organization performing its mission to parameter variations.

Assume that process Pi is preceded in the process graph by processes $P_j,\ldots,P_k$ and that the results of process $P_i$ are needed by processes $P_m,\ldots,P_n$. In addition assume that a different process $P'_i$ is preceded by processes $P'_j,\ldots,P'_k$ and its results are needed by processes $P'_m,\ldots,P'_n$. Figures 3-1 and 3-2 show the partial precedence process graphs for both cases. Finally, it is assumed that resource Rp must perform both processes $P_i$ and $P'_i$.

If resource $R_p$ can perform the two processes in any order (i.e., finish $P_i$ first and then $P'_i$ or vice versa), then the two processes have to "compete" for the resource and the Petri net model is given by Fig. 3-1.

The resource $R_p$ is modeled as a place, initially with one token, and the processes $P_i$ and $P'_i$ are modeled as transitions. The completion of both process $P_i$ and $P'_i$ require the availability of resource $R_p$ and the completion of all the corresponding preceding processes in their process graphs. More specifically, when process $P_j$ is completed a token will be placed in the place labeled "from process $P_j$". It is clear that if all the processes $(P_j,\ldots,P_k)$ are completed and the resource $R_p$ is available (token present in the "resource $R_p$" place) then the transition "process $P_i$ " will fire, and after a delay of $t_{ip}$ (processing time of resource $R_p$ for process $P_i$)

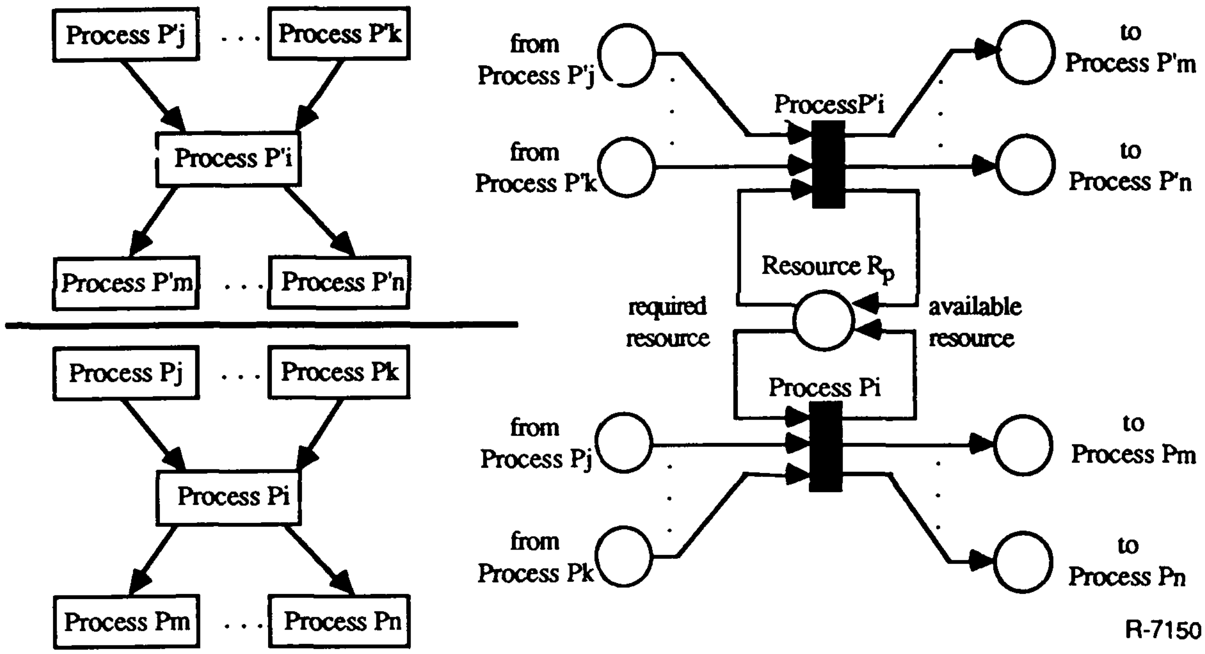



Figure 3-1. Petri Net Representation of $R_p$ Performing $P_i$ and $P'_i$ in any Order.

a token will be placed in each of the output places and the resource $R_p$ will be available again. Similarly, if all the processes ($P'_j,..,P'_k$) are completed and the resource $R_p$ is available, then the transition "process $P'_i$ " will fire, and after a delay of $t_{ip'}$, a token will be placed in each of the output places and the resource $R_p$ will be available again. Since both processes $P_i$ and $P'_i$ need the same resource $R_p$, the first process with all the preceding processes finished will use the resource $R_p$ first. If there is a tie, then the process with the highest priority will be performed first.

For the case where resource $R_p$ has to perform process $P_i$ first and then performs process $P'_i$, the Petri net model is given in Fig. 3-2. In this case resource $R_p$ must perform $P_i$ first, and then when process $P_i$ is completed its transition in the model fires and releases $R_p$ to be used by process $P'_i$. In this case even if all the preceding processes ($P'_j,...P'_k$) are completed, if process $P_i$ is not finished process $P'_i$ cannot start. This is the same as introducing an additional precedence constraint for process $P'_i$, not included in the process graph.

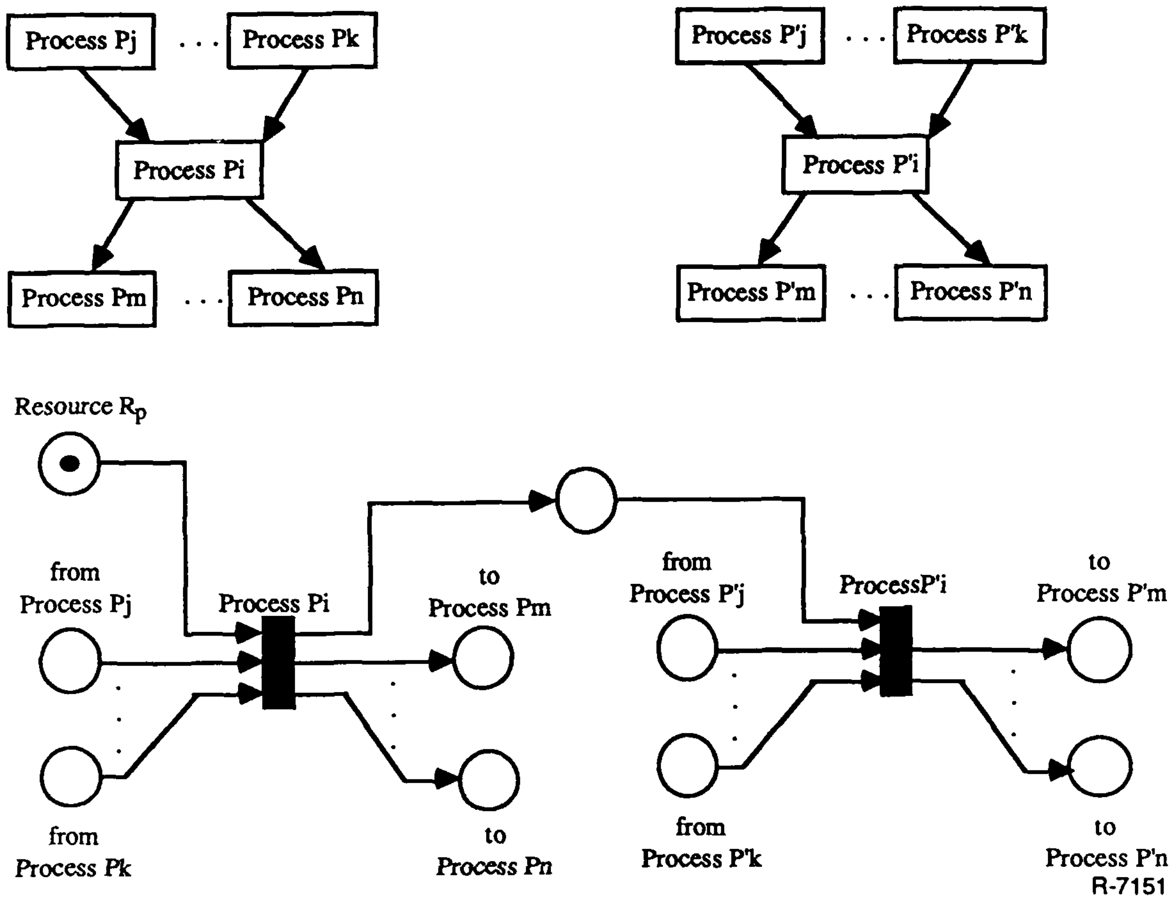



Figure 3-2. Petri Net Representation of $R_p$ Performing $P_i$ First and then $P'_i$.

Next we introduce two generic models for utilizing communication links to transmit the necessary information from one process to the others. For the development of the generic models we assume that process $P_i$ (performed by $R_p$) uses the communication link p-q (from resource $R_p$ to resource $R_q$) to transmit information to process $P_j$ (performed by $R_q$), and that process $P_k$ (performed by $R_p$) uses the same link (link p-q) to transmit information to process $P_l$ (performed by $R_q$). Fig. 3-3a shows a model of the communication link usage when both processes can transmit information in any order (the link is used first by process $P_i$ and then by process $P_k$, or vice versa). When process $P_i$ is completed a communication delay equal to $v_{ij}/\mu_{pq}$ is necessary to transmit the results via link p-q, and when process $P_k$ is completed the communication delay for using the link p-q is $v_{kl}/\mu_{pq}$.

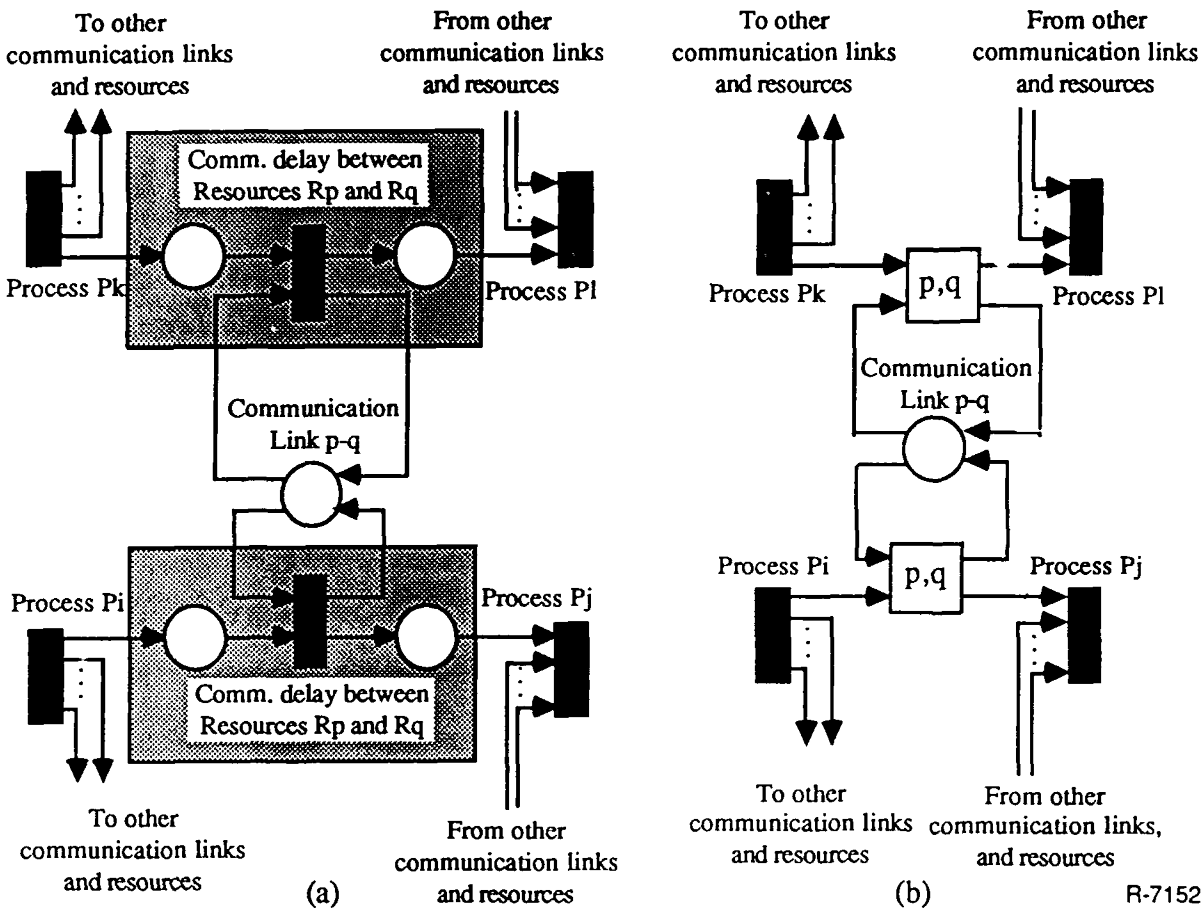



Figure 3-3. Petri Net of the Communication Link with no Prescribed Order.

Each communication delay is modeled as a transition (with the corresponding communication delay) that has as inputs the token coming from the completion of the corresponding process ($P_i$ or $P_k$) and the token representing the availability of the communication link p-q. Note that the link p-q is shared and it will not be available at all times for both processes. For convenience the shaded region in Fig. 3-3a will be replaced by a box with "p,q" representing the communication link p-q as shown in Fig. 3-3b.

Figure 3-4 shows the model of the communication when process $P_i$ must use the link first. In this case the two processes ($P_i$ and $P_k$) do not "compete" for the communication link. When process $P_i$ is finished using the link p-q, a token is generated indicating that link p-q is available for process $P_k$.

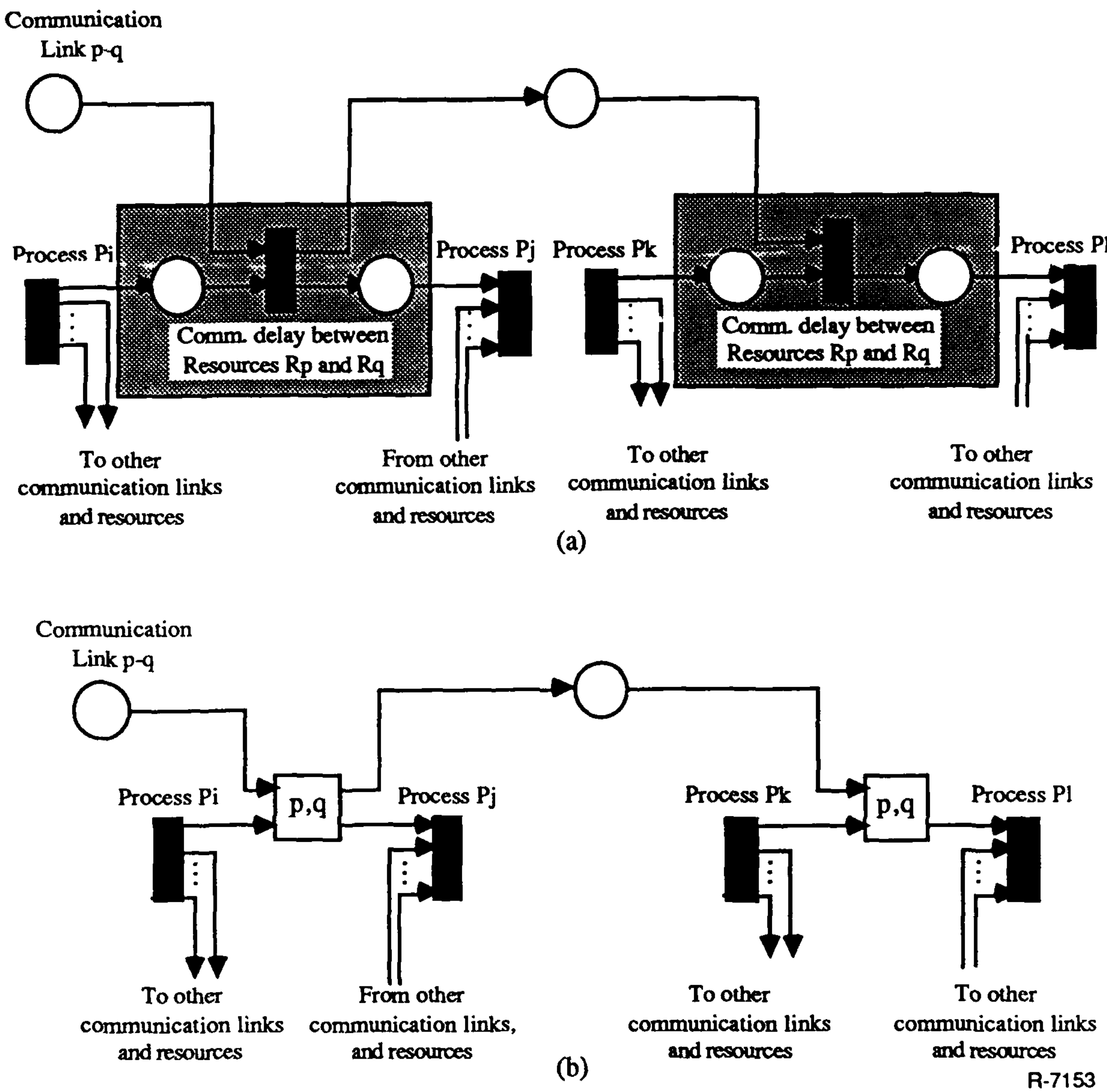



Figure 3-4. Petri Net of the Communication Link when $P_i$ Must Use it Before $P_l$.

In subsection 2.3 we introduced Mapper for finding the optimal or near-optimal mapping of processes onto resources in $C^3$ organizations. In this section we use Petri nets to analyze the sensitivity of the solution by simulating the organization performing the mission (sequence of processes). The idea is to model the organization structure that completes the processes and then to vary certain parameters and analyze the effects of the variation on the performance of the system.

To do the mapping we also used the following information:

a) Time required for each resource to complete each process

b) Amount of information transmitted from process to process

c) Communication rates between resources

As stated previously, the completion of a process by a resource is modeled as a transition, with its delay being the time required by that resource to finish the process. One can combine the amount of information transmitted from process to process and the communication rates to find the communication time between resources for a specific process assignment. This communication can then be modeled as a transition whose delay is the communication time required by the resources performing the processes.

The effects of errors in the estimates of the processing time required for each resource to complete each process and the communication delay between resources may be analyzed by changing these values, simulating the perturbed system, and examining the new completion time of the final process and the new resource utilizations. An example of the process is given below.

## 3.2 EXAMPLE

Consider the simple process sequence and organization structure shown in Fig. 3-5.

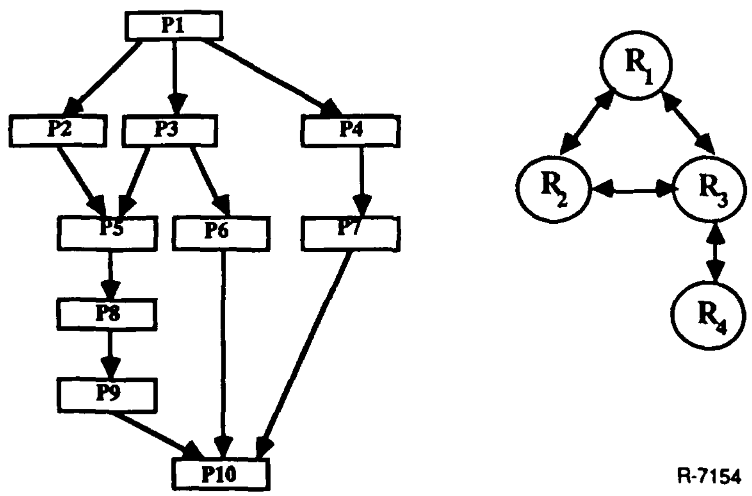



Figure 3-5. Process Graph and Resource Graph of an Example.

Tables 3-1 through 3-3 identify all the characteristics of the processes and the resources, which are used to obtain the constraints for the process mapping problem. Specifically, the processes and resources for this example have the following characteristics:

PC1: The graph of processes $G_p = (V_p, E_p)$ with precedence constraints is given in Fig. 3-5.

PC2: The amount of information ($v_{ij}$) transmitted from process $P_i$ to $P_j$ when the transmission exists is given in Table 3-1 (in number of messages).

PC3: The number of resources ($n_i$) required to complete process $P_i$ is $n_i$=1 for all processes.

PC4: *The process difficulty ($d_i$) is assumed to be the same for all processes and,* therefore, $d_i$=1 $\forall$i.

PC5: There is no consistency/continuity requirement in this example.

OC1: The graph of resources $G_r = (V_r, E_r)$ with precedence constraints is given in Fig. 3-5.

OC2: The time ($t_{ip}$) required for each resource $R_p$ to complete each process $P_i$ is given in Table 3-2.

OC3: The communication rates ($\mu_{pq}$) between resources for each resource $R_p$ and resource $R_q$ are given in Table 3-3.

OC4: The work load capacity ($L_p$) for this example is assumed to be $L_p$ = 10 for all the resources.

We use the algorithm of subsection 2.3 to minimize the completion time of the last process (in our case $P_{10}$) under the same constraints described in subsection 2.2:

$$\min_{R_p \in V_p} (s_{10p} + t_{10p}) \tag{3-11}$$

Figure 3-6 shows the resulting use of the communication links and the process-to-resource assignments. For example, resource $R_2$ completes $P_1$ (0-10sec.) and then the same resource completes $P_4$ (10-40sec.). Note that although $P_4$ requires data from $P_1$, there is no communication delay since $R_2$ is performing both processes. The result of $P_4$ has to be transmitted from $R_2$ to $R_1$ (40-50sec.) so that resource $R_1$ can start processing $P_7$. At the end, resource $R_1$ has to complete the final process $P_{10}$; processes $P_9$ and $P_6$ precede $P_{10}$, and before

## TABLE 3-1. AMOUNT OF DATA TO BE TRANSMITTED FROM PROCESS TO PROCESS

Number of messages transmitted to process

Number of messages transmitted from process

| | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 |
|---|---|---|---|---|---|---|---|---|---|---|
| P1 | | 5 | 5 | 10 | | | | | | |
| P2 | | | | | 5 | | | | | |
| P3 | | | | | 5 | 5 | | | | |
| P4 | | | | | | | 10 | | | |
| P5 | | | | | | | | 10 | | |
| P6 | | | | | | | | | | 10 |
| P7 | | | | | | | | | | 10 |
| P8 | | | | | | | | | 10 | |
| P9 | | | | | | | | | | 10 |
| P10 | | | | | | | | | | |

R-7195

## TABLE 3-2. TIME REQUIRED BY RESOURCES TO COMPLETE EACH PROCESS (SEC)

| | Resource R1 | Resource R2 | Resource R3 | Resource R4 |
|---|---|---|---|---|
| Process P1 | 20 | 10 | 10 | 10 |
| Process P2 | 40 | 20 | 20 | 20 |
| Process P3 | 26 | 52 | 26 | 26 |
| Process P4 | 30 | 30 | 30 | 30 |
| Process P5 | 10 | 10 | 10 | 10 |
| Process P6 | 50 | 100 | 100 | 50 |
| Process P7 | 44 | 88 | 88 | 44 |
| Process P8 | 15 | 15 | 15 | 15 |
| Process P9 | 10 | 10 | 20 | 20 |
| Process P10 | 10 | 10 | 20 | 20 |

R-7196

## TABLE 3-3. COMMUNICATION RATE FOR LINKS

Messages/sec

| | |
|---|---|
| Link R1-R2 | 1 |
| Link R1-R3 | 2 |
| Link R2-R3 | 2 |
| Link R3-R4 | 2 |

R-7197

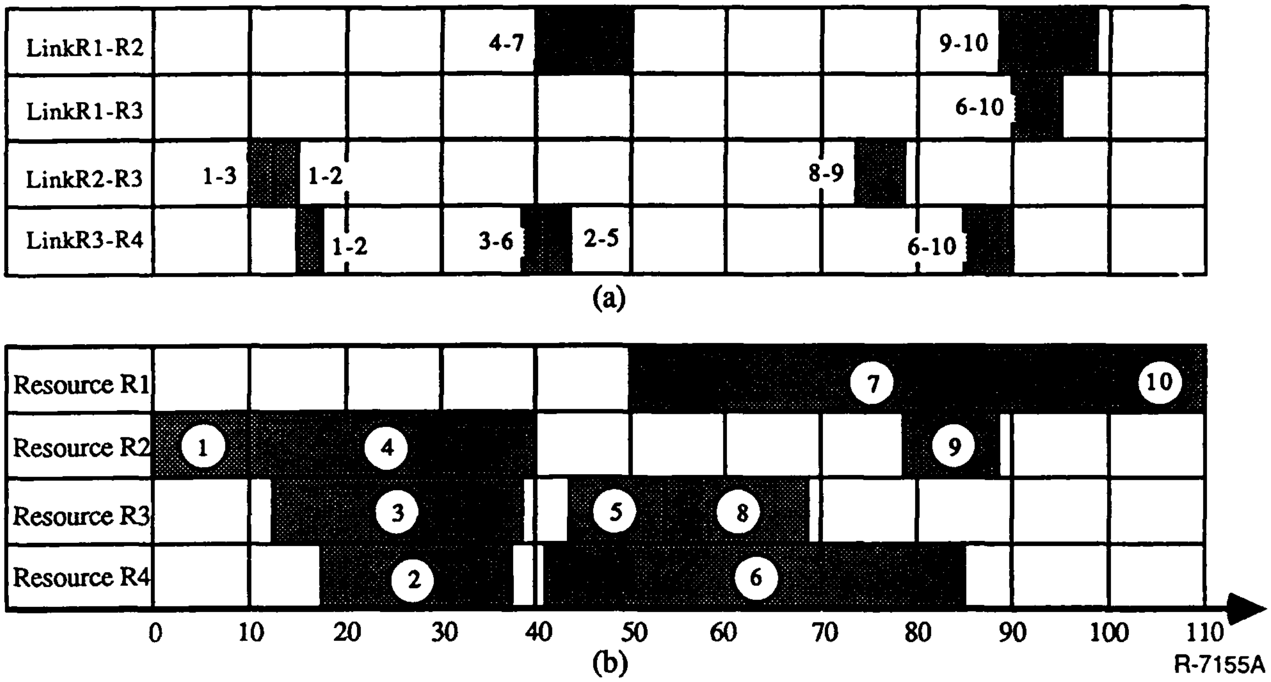



Figure 3-6. Communication Link Use and Resource Allocation.

$P_{10}$ starts the results of $P_6$ and $P_9$ are communicated to $R_1$ via the links $R_1$-$R_2$ and $R_1$-$R_3$-$R_4$. Note that there is not a direct link between $R_4$ and $R_1$; that is why the path $R_1$-$R_3$-$R_4$ was used to transmit the results of $P_6$ from $R_4$ to $R_1$. One can see that the ($P_1$-$P_4$-$P_7$-$P_{10}$) path is a critical one because any small variation in the process completion and communication delays in the path will cause the final time to vary by the same amount.

In order to simulate the organization performing the processes according to the process-resource assignment, a Petri net has to be constructed. The generic models introduced in subsection 3.1 will be used to construct two different Petri nets. The first step for both Petri nets is to construct the sequence of processes as described in the process graph with the communication links (see Fig. 3-7). (The boxes representing communication links in Fig. 3-7 were defined in subsection 3.1.)

In the first Petri net we will model the solution of the process mapping problem preserving the order of completion of the processes and the order of usage of the communication links. That is to say that for this specific example resource $R_1$ will first complete process $P_7$

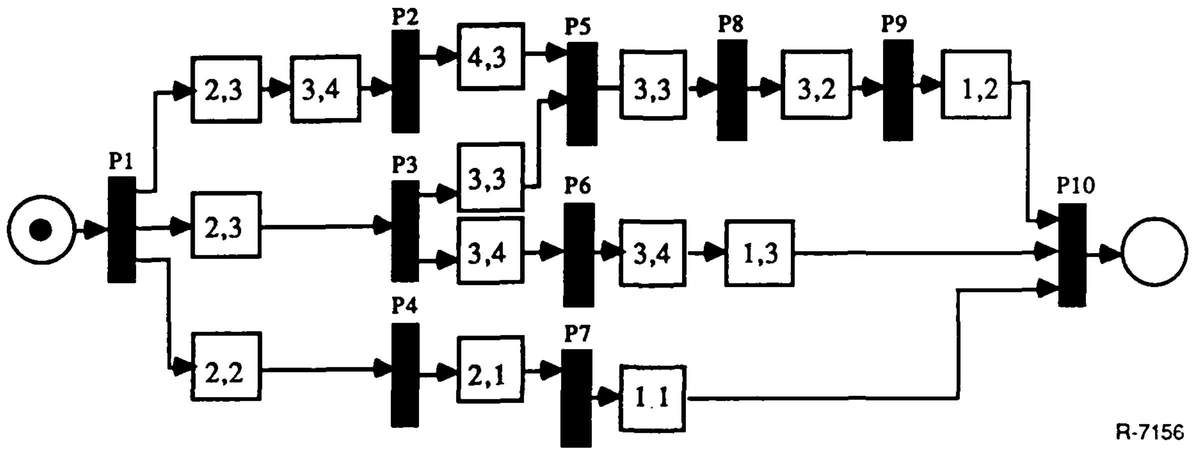


Figure 3-7. Petri Net Representation of the Process Graph.

and then process $P_{10}$, similarly resource $R_2$ will first complete process $P_1$, then process $P_4$, etc. The communication link 1-2 (from resource $R_1$ to resource $R_2$) will transmit the results of process $P_4$ to process $P_7$ first and then it will transmit the results of process $P_9$ to process $P_{10}$. In similar fashion the completion order of the processes and the order of the usage of the links is preserved. Figure 3-8 shows the complete Petri net for this case. Note that the Petri net in Fig. 3-8 can be generated automatically from the Mapper solution.

Using the Petri net of Fig. 3-8 one can determine the sensitivity of the organization to variations in different parameters. For example, in Fig. 3-9a the change in completion time $s_{t1}+t_{t1}$ (note that since $P_{10}$ is the terminal process, $s_{t1}=s_{10,1}$ and $t_{t1}=t_{10,1}$) is found by varying the time required for resource $R_4$ to complete process $P_2$. One can see that nominally the path ($P_1$-$P_2$-$P_5$-$P_8$-$P_9$-$P_{10}$) in the process graph is not the critical one. When the variation of the time necessary for resource $R_4$ to complete $P_2$ ($t_{2,4}$) is greater than 7.5, the same path becomes the critical one and consequently the completion time becomes linear to the $t_{2,4}$ variation. Similarly in Fig. 3-9b the completion time ($s_{t1}+t_{t1}$) is found by varying the time required for resource $R_2$ to complete process $P_4$ ($\Delta t_{4,2}$). Nominally, the path ($P_1$-$P_4$-$P_7$-$P_{10}$) is the critical path and the variation of the completion time of the final process $s_{t1}+t_{t1}$ is linear in the variation in $t_{4,2}$. If $\Delta t_{4,2}$ is decreased, there is a point where the same path is no longer critical, and the $\Delta t_{4,2}$ no longer affects the completion time $s_{t1}+t_{t1}$.

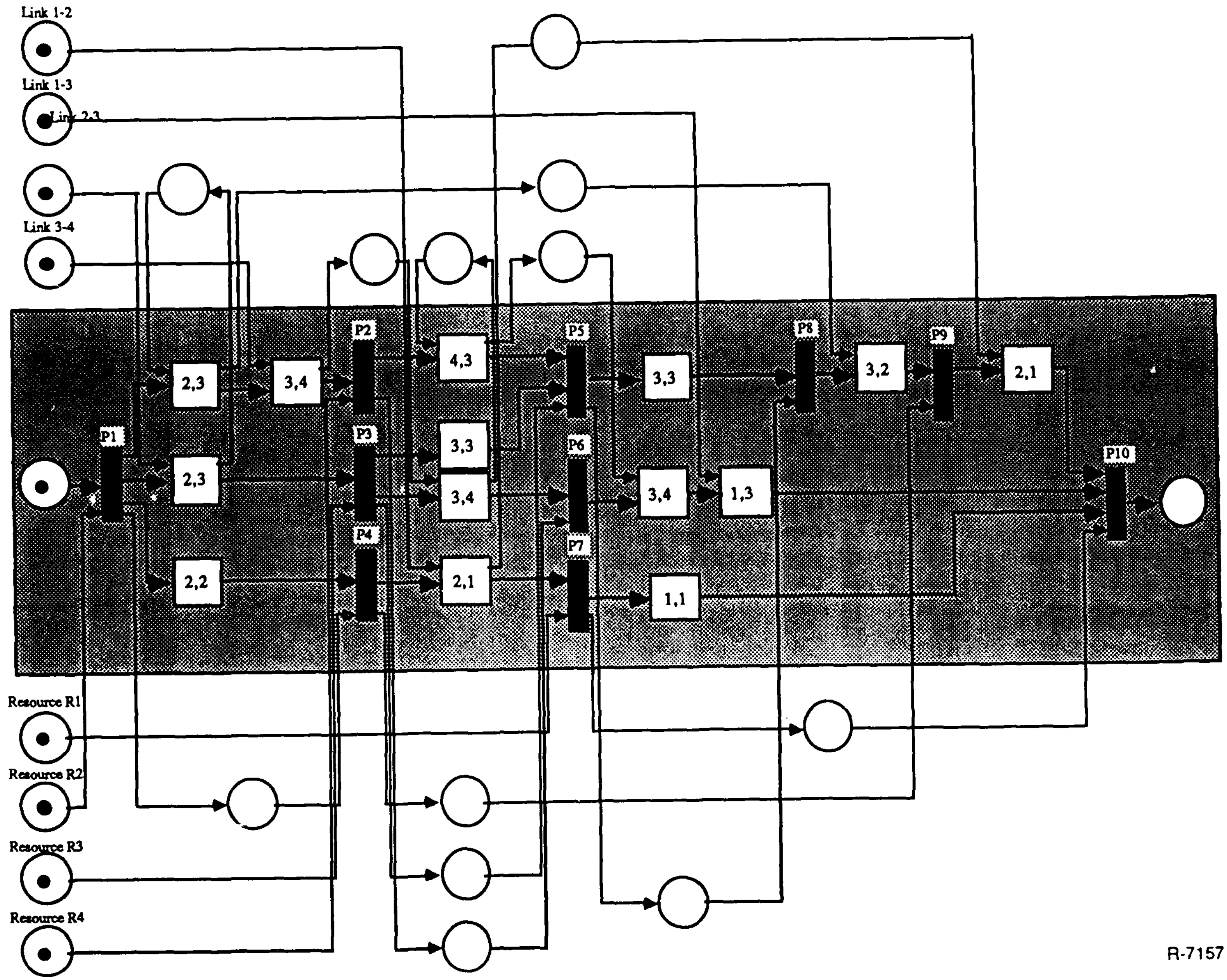
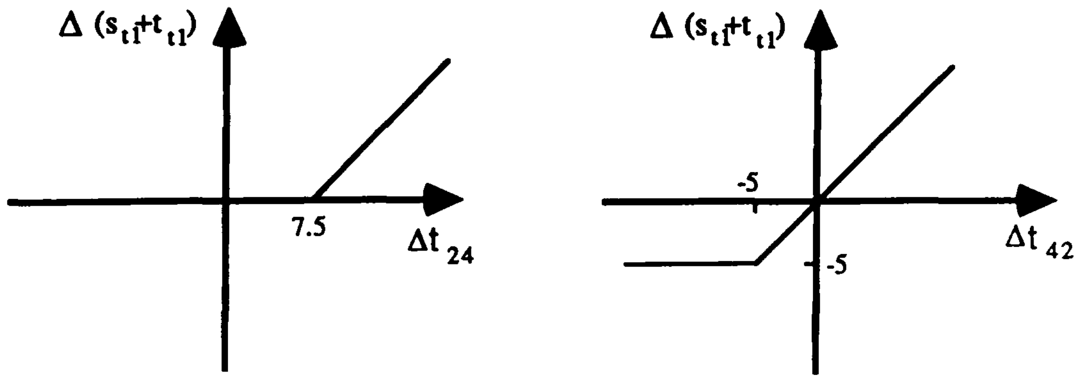


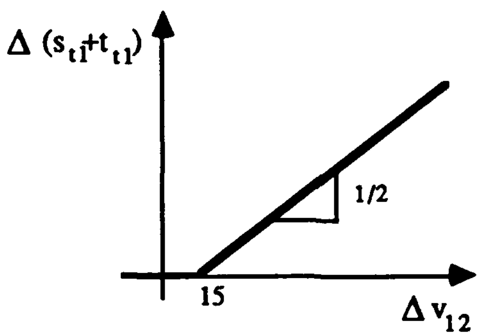

Figure 3-8. Petri Net with Preserved Sequences.




(a) (b) R-7158

Figure 3-9. Sensitivity to Completion Time.

In Fig. 3-10 the completion time of the final process is found for different amounts of communication $v_{12}$ (number of messages) transmitted from process $P_1$ to process $P_2$. It is clear that processes $P_1$ and $P_2$ are not in the critical path until the required communication $v_{12}$ is increased by 15 messages. Then the path with processes $P_1$ and $P_2$ becomes critical and the increase in $v_{12}$ causes the final time to increase accordingly.




R-7159

Figure 3-10. Sensitivity to Message Number.

In the second Petri net model the order of completion of the processes and the order of the communication usage of the links are not nessessarily preserved. The resources and the communication links are used in a first-come, first-served basis. In the specific example resource $R_4$ still completes process $P_2$ and $P_6$, but not necessarily in the order given by the solution from the resource allocation algorithm. Figure 3-11 shows the second Petri net model.

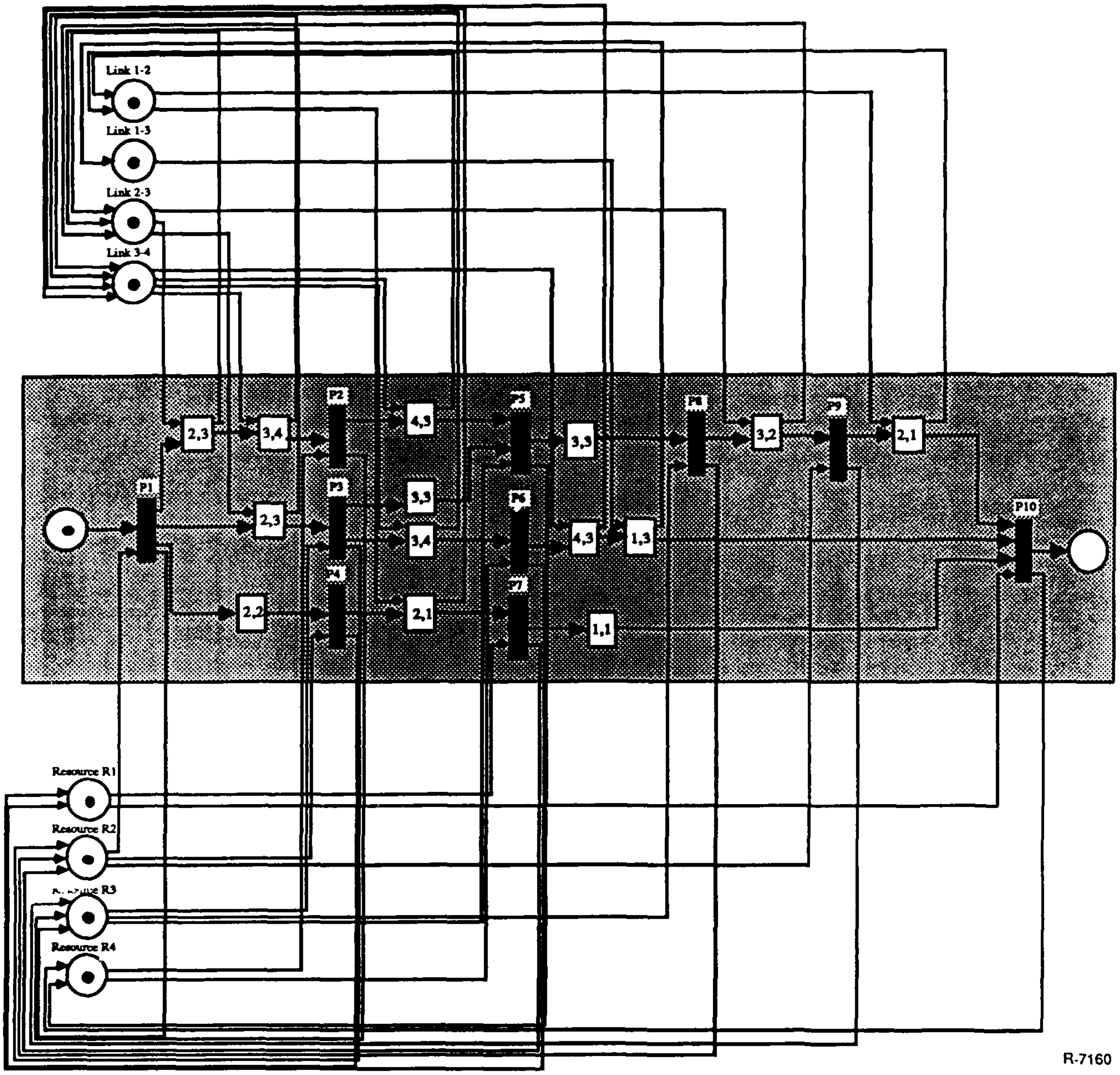



Figure 3-11. Petri Net without Preserved Sequences.

With the Petri net in Fig. 3-11, if the communication and processing delays of the resources are changed, the order of execution of the processes may also change; this is true because the availability of the resources and the communication links will differ. Note that the Petri net in Fig. 3-11 can also be generated automatically from the solution of the resource allocation problem.

In Fig. 3-12 the completion time of the final process is found for different amounts of communication $v_{12}$ (number of messages) transmitted from process $P_1$ to process $P_2$; (compare Fig. 3-12 with Fig. 3-10). The discontinuity in the plot of Fig. 3-12 is due to the fact that for low $v_{12}$ resource $R_4$ is operating on process $P_2$ before it starts operating on process $P_6$. In contrast, when $v_{12}$ is increased by more than 47 messages, process $P_6$ is completed first. Note that process $P_2$ and process $P_6$ can be performed in parallel because there is not a precedence constraint between them. Therefore, the discontinuity indicates that for large $v_{12}$ a reconfiguration in the organization (i.e., process reassignment) may be necessary. If $v_{12}$ is stochastic with wide variations (large standard deviation), one may want to change the resource mapping to one that will be more robust to these kinds of variations (i.e., $P_2$ and $P_6$ to be performed by different resources).

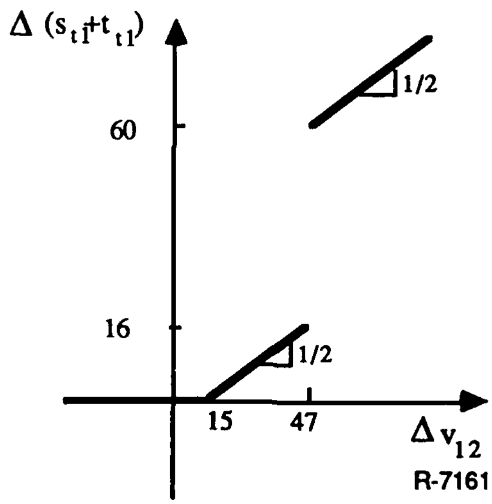



Figure 3-12. Sensitivity to Message Number — Without Preserving Sequences.

Figure 3-13 shows the optimal process mapping for the case where the amount of communication between process $P_1$ and process $P_2$ is increased by 50 messages ($v_{12} = 55$). One can see that process $P_2$ is now completed by resource $R_1$, which also completes process $P_1$. Consequently, the communication cost to transmit information between $P_1$ and $P_2$ is zero

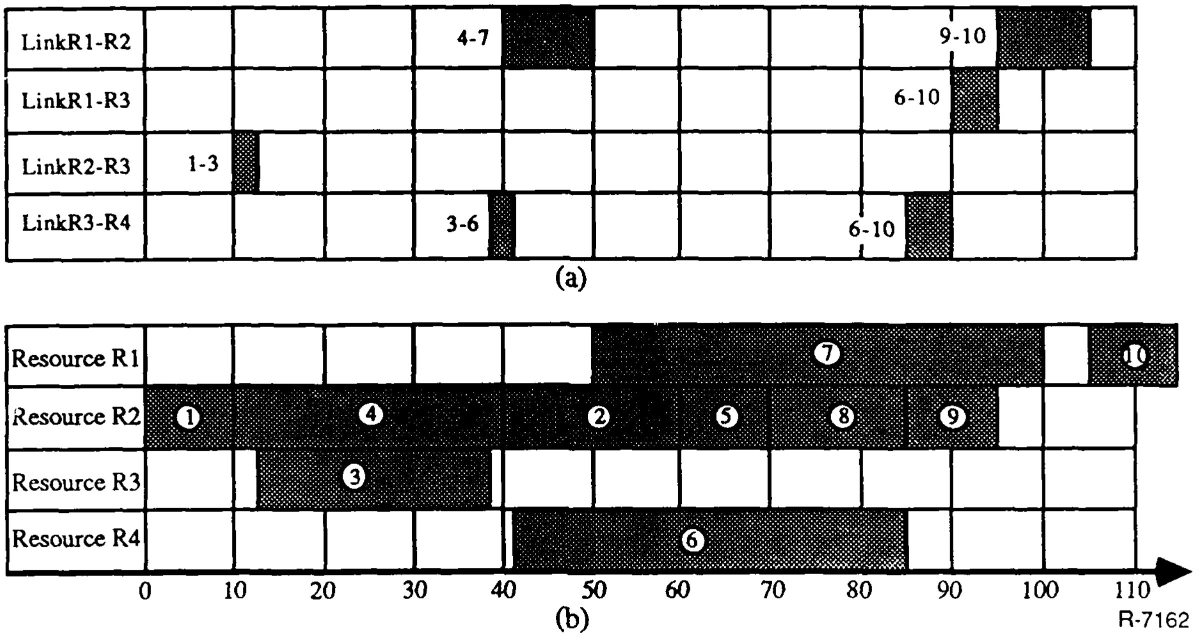



Figure 3-13. Communication Link Use and Resource Allocation — Increased Communication.

because no communication link is utilized. A wide variation in $v_{12}$ will not cause catastrophic delays as in the previous case.

Note that the completion time (115 sec.) of the final process for the new solution as shown in Fig. 3-13 is approximately 4.55 percent greater than the completion time (110 sec.) of the original solution (shown in Fig. 3-6). For 4.55 percent deterioration in nominal performance, the new solution is robust to wide variations in the amount of information ($v_{12}$) transmitted between processes $P_1$ and $P_2$. If Fig. 3-13a is compared with Fig. 3-6a, it is clear that the greater communication cost ($\Delta v_{12} = 50$) caused fewer communications and, therefore, more centralized processing (i.e., resource $R_2$ completes $P_1$, $P_2$, $P_4$, $P_5$, $P_8$, $P_9$).

## SECTION 4

## CONCLUSION AND FUTURE RESEARCH

### 4.1 CONCLUDING REMARKS

In this report we have developed the Mapper algorithm to map missions onto organizations consisting of resources of varying capability, communication connectivity, and communication channel capacity. The objective is to minimize the completion time of the mission without violating the constraints that are defined in terms of the requirements of the processes and the capabilities of the resources.

The organization performing the mission, with the process mapping specified by Mapper, is modelled using Petri nets. The model can be generated automati lly from the definition of the organizational structure and the Mapper solution. Then simulations can be performed examining the sensitivity of the overall performance to changes in the parameters of the mission and organization.

To illustrate the results two examples were used. In the first example a Control and Reporting Center (CRC) was used to complete a mission with 37 processes. A second more simple example illustrated both the resource allocation solution and the modeling and analysis of the organization with Petri nets.

### 4.2 RESEARCH ISSUES

Usually in $C^3$ organizations there is a tradeoff between the time required for an organizational element to complete a task and the quality of its performance. In the formulation of the process mapping problem given above, the objective was to minimize the completion time of the final process without taking into account the quality of the performance of each resource.

More research is needed to incorporate the performance error for each resource. One approach is to define for each resource-process ($R_q$-$P_i$) pair a performance curve similar to the

curves shown in Fig. 4-1. In essence the error curve is a measure of the quality of the performance of the resource when it operates on a process. These curves can be derived from the capabilities of the resource. Future work should also address developing error curves for multiple resources performing a process, where phenomena such as diminishing returns and synergism can be represented.

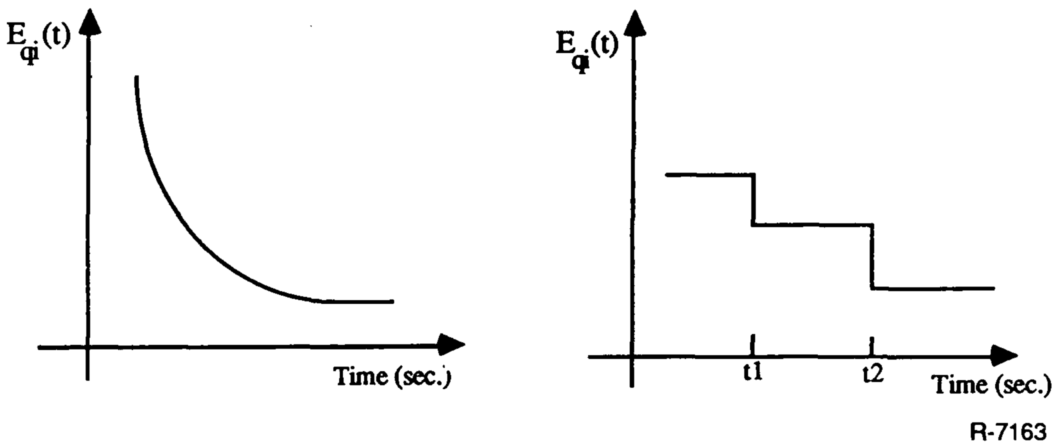



Figure 4-1. Typical Performance Error for Resources Performing in a Process.

Having performance error curves for each resource-process pair, one can formulate the process mapping problem using a criterion such as the following:

a) Minimize the product of the overall error (a weighted sum of the individual errors) and the completion time, or

b) Minimize completion time such that the overall error is bounded by an a priori defined upper limit, or

c) Minimize the completion time such that the overall error is bounded by an a priori defined upper limit and certain processes have to be executed with specific accuracies.

Currently Mapper assumes that the resources can communicate among themselves and perform processes at the same time (in parallel). For many resources this is true (i.e., ships, tanks, airplanes, computers, etc.) but for others (i.e., humans) the assumption may not be valid. We must augment the algorithm with the flexibility of defining whether the resource can communicate and operate on missions in parallel or only in series.

Another research issue is enhancing the flexibility of the workload capacity constraint. Specifically, each resource should be allocated to minimize the completion time of the mission. In case there is more than one resource that can perform a process with the same completion time, the resource with more free time (slack time) should be chosen to perform the process. In this way we are distributing the resources in a better, more uniform way. In fact, the algorithm should provide enough flexibility so that one can choose if it is desired to trade off better distribution of the resources for larger completion time.

The technique should be generalized to address the performance of multiple missions by an organization and priorities among those missions or mission elements. The dependence of task difficulty, and perhaps task number, upon aspects of the external environment (such as number of aircraft in an enemy raid) should also be explicitly incorporated. As mentioned in subsection 2.2, Mapper currently allows only one type of time overlap (at the start) for multiple resources performing a process. It may be useful to allow other forms of overlap as well.

Finally, some consideration should be given to a technique for "closing the loop" in the mapping process or, for that matter, the overall organization design process illustrated in Fig. 1-1. The sensitivity analysis technique we have presented is manual, and an automated technique is needed for searching the parameter spaces for sensitivities, and transforming those sensitivities into improvements in process assignments or organizational changes.

## REFERENCES


1. Wohl, J.G., and R.R. Tenney, "Integrated Analysis Techniques for Command, Control, and Communications Systems," Volume I: Methodology, AAMRL-TR-87-040, Harry G. Armstrong Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, Ohio, November 1987.

2. Serfaty, D.S., et al., "Automation Requirements Definition and Specification for Integrated Analysis Techniques (AutoIAT)," Technical Report TR-315, ALPHATECH, Inc., Burlington, Massachusetts, November 1986.

3. Moore, K.E., F. Genoese-Zerbi, and D.L. Saisi, "IDEF ALPHA User's Manual, Technical Report TR-358-1, ALPHATECH, Inc., Burlington, Massachusetts, April 1988.

4. Wohl, J.G., "Representing the Organization Dimension in Petri Net Form," Technical Report TR-350, ALPHATECH, Inc., Burlington, Massachusetts, September 1987.

5. Lee, R.T., K.R. Pattipati, and P.B. Luh, "On the Algorithm-Architecture Mapping Problem," submitted to IEEE Transactions on Computers, July 1988.

6. Bokhari, S.H., "On the Mapping Problem," IEEE Trans. on Computers, Vol. C-30, March 1981, pp. 207-214.

7. Ganey, M.R., and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, New York, 1979.

8. Horowitz, E., and S. Sahni, Algorithms: Design and Analysis, Computer Science Press, Potomac, MD, 1978.

9. Karp, R.M., "Reducibility Among Combinatorial Problems," in R.E. Miller and J.W. Thatcher (Eds), Complexity of Computer Compilations, Plenum Press, New York, pp. 85-103.

10. Hu, T.C., "Parallel Sequencing and Assembly Line Problem," Operations Research, Vol. 9, November 1961, pp. 841-848.

11. Sethi, R., "Scheduling Graphs on Two Processors," SIAM Journal on Computing, Vol. 5, 1975, pp. 73-82.

12. "Tactical Air Control System (TACS) Surveillance and Control Of Tactical Air Operations," TACR 55-44 (C1), Department of The Air Force, Headquarters Tactical Air Command, Langley Air Force Base, Virginia, 27 January 1978.

## SYMBOLS

| | |
|---|---|
| $A_i$ | Level of process i |
| $C_t$ | Completion time of process t |
| $d_i$ | The difficulty of process i |
| $D_{iq}$ | Time at which data from all parents of process i become available at resource q |
| $\in$ | An element of |
| $E_p$ | Set of ordered process pairs denothing data dependencies |
| $E_r$ | Set of unordered resource pairs denoting communication links |
| $F_i$ | Set of feasible resources to which process i can be assigned without violating the workload capacity constraint |
| G | Goal |
| $g_i$ | The $i^{th}$ consistency/continuity constraint — a group of processes to be completed by a single resource |
| $L_p$ | Workload capacity of resource p |
| M | Number of resources |
| N | Number of processes |
| NG | Number of specified groups of processes |
| $n_i$ | The number of resources required to complete process i (all resources start at the same time) |
| O | Organizational element |
| $P_i$ | Process i |
| $Q_p$ | Sequence of ordered pairs representing process sequence performed by resource p |
| $R_q$ | Resource q |
| $s_{ip}$ | Time that resource p starts on process i |
| $t_{ip}$ | Time required for resource p to complete process i |
| $v_{ij}$ | Amount of information that must be transmitted from process i to process j |
| $V_p$ | The set of all processes in the mission |
| $V_r$ | The set of all resources in the organization |
| $\mu_{pq}$ | Communication rate between resources p and q |
| $\forall$ | For all |
| $\exists$ | There exists |



☆U.S. Government Printing Office: 1990 — 748-002/00248